KEPLA: A Knowledge-Enhanced Deep Learning Framework for Accurate Protein-Ligand Binding Affinity Prediction

Han Liu, Keyan Ding, Peilin Chen, Yinwei Wei, Liqiang Nie, *Senior Member, IEEE*, Dapeng Wu, *Fellow, IEEE*, and Shiqi Wang, *Senior Member, IEEE*

arXiv:2506.13196v4 [cs.LG] 22 Jan 2026

***Abstract*—Accurate prediction of protein-ligand binding affinity is critical for drug discovery. While recent deep learning approaches have demonstrated promising results, they often rely solely on structural features of proteins and ligands, overlooking their valuable biochemical knowledge associated with binding affinity. To address this limitation, we propose KEPLA, a novel deep learning framework that explicitly integrates prior knowledge from Gene Ontology and ligand properties to enhance prediction performance. KEPLA takes protein sequences and ligand molecular graphs as input and optimizes two complementary objectives: (1) aligning global representations with knowledge graph relations to capture domain-specific biochemical insights, and (2) leveraging cross attention between local representations to construct fine-grained joint embeddings for prediction. Experiments on two benchmark datasets across both in-domain and cross-domain scenarios demonstrate that KEPLA consistently outperforms state-of-the-art baselines. Furthermore, interpretability analyses based on knowledge graph relations and cross attention maps provide valuable insights into the underlying predictive mechanisms. The code and datasets of KEPLA are available at the GitHub repository (https://github.com/hanliu95/KEPLA).**

***Index Terms*—Protein-ligand binding affinity, knowledge graph, cross attention, virtual screening, interpretability.**

I. Introduction

THE prediction of protein-ligand binding affinity (PLA) is a key task in computational drug discovery [1]. Ligands, including small molecules and biologics, can bind to proteins as agonists or inhibitors. Their interaction strength is measured by binding affinity, which reflects how effectively a ligand can modulate a target protein. Although experimental assays can measure binding affinity, they are labor-intensive and time-consuming. As a result, in silico approaches have attracted increasing attention [2]. These methods enable fast affinity prediction for early-stage drug discovery, facilitating candidate ranking while reducing reliance on costly experimental screening [3].

(Corresponding author: Shiqi Wang.)

Han Liu, Peilin Chen, Dapeng Wu, and Shiqi Wang are with the Department of Computer Science, City University of Hong Kong, Hong Kong (SAR), China (e-mail: hanliu.sdu@gmail.com; plchen3@cityu.edu.hk; dapengwu@cityu.edu.hk; shiqwang@cityu.edu.hk).

Keyan Ding is with the ZJU-Hangzhou Global Scientific and Technological Innovation Center, Zhejiang University, Hangzhou, China (e-mail: dingkeyan@zju.edu.cn).

Yinwei Wei is with the School of Software, Shandong University, Jinan, China (e-mail: weiyinwei@hotmail.com).

Liqiang Nie is with the College of Informatics, Harbin Institute of Technology (Shenzhen), Shenzhen, China (e-mail: nieliqiang@gmail.com).

Recently, along with their success in other areas, deep learning approaches have rapidly progressed for computational PLA prediction [4]. The scientific community has primarily divided these deep learning approaches into interaction-based and interaction-free methodologies, which are distinguished by whether they rely on physical interactions to inform predictions. Interaction-based approaches make predictions based on the known three-dimensional (3D) structures of complexes and the physical interactions between proteins and ligands [5]–[7]. As the number of biological targets with available 3D structures is limited, interaction-based approaches are less useful than interaction-free ones when faced with unknown protein-ligand complexes. In contrast, without relying on direct physical interactions, interaction-free approaches infer binding affinity from protein and ligand lower-dimensional data, such as amino acid sequence and simplified molecular input line entry system (SMILES) string [8]. Regarding PLA prediction as a regression task, interaction-free approaches first feed the protein and ligand inputs into different encoders such as convolutional neural network (CNN) [9], [10], graph neural network (GNN) [11], [12], and Transformer [13]–[15]. With these advanced deep learning techniques, data-driven representations can be automatically learned from large-scale protein and ligand data. Subsequently, the binding affinities between proteins and ligands are predicted through a decoder that incorporates and processes their representations. Obviously, interaction-free methods have wider applications, as they take 1D or 2D structural information from proteins and ligands as input and thus have fewer restrictions on data.

Despite their promising results, existing interaction-free models still suffer from two key limitations: (1) their reliance solely on structural information creates performance bottlenecks, and (2) their predictions often lack scientific interpretability. To overcome these limitations, we argue that interaction-free models should incorporate biochemical factual knowledge during the encoding of proteins and ligands. Such knowledge provides rich external information that not only enhances predictive accuracy but also offers scientific grounding for interpretability. For proteins, Gene Ontology (GO) [16] annotations describe their molecular functions, biological processes, and cellular components, which are highly relevant to protein-ligand binding. For instance, if a protein's molecular function includes *ATP binding*, it may exhibit high affinity for ATP-like ligands. Involvement in biological processes such as *signal transduction* can suggest potential molecu-

lar interactions or pathway associations. Similarly, cellular component annotations like *cell membrane* can inform ligand accessibility. For ligands, knowledge can be derived from molecular descriptors and chemical features. Ligand properties (LP) such as the number of *hydrogen bond donors* (e.g., amino groups, phenolic hydroxyl groups) and *acceptors* (e.g., oxygen, nitrogen atoms) are crucial for forming hydrogen bonds with proteins. These interactions play a pivotal role in protein-ligand binding, making such properties essential for accurate PLA prediction.

However, integrating such domain knowledge presents two key challenges: (1) how to organize the diverse and complex knowledge into a form that the model can easily process, and (2) how to enable the model to sufficiently capture the organized knowledge. To address the first challenge, knowledge graphs (KGs) [17] are expected to be the optimal solution due to their proven capabilities in knowledge tasks. KGs constructed from GO and LP can systematically describe biochemical knowledge of proteins and ligands via the form of *entity-relation-entity* triples. Correspondingly, KG embedding techniques [18] can map proteins, ligands, and related entities into continuous vector spaces, preserving both semantic and relational information. For the second challenge, a straightforward approach is to directly use the KG-derived embeddings of proteins and ligands, alongside structural representations, for PLA prediction. However, this approach falls short for knowledge-sparse entities, as KG embeddings cannot substitute structural encoders when only amino acid sequences or SMILES strings are available. Therefore, incorporating the KG to enhance the structural information encoder is a more effective strategy that can offer the best of both worlds. To this end, we propose a joint learning framework that bridges structural encoding and KG embedding through multi-objective learning, ensuring that biochemical knowledge is effectively injected during the representation learning process.

In this paper, we propose a knowledge-enhanced protein-ligand binding affinity prediction (KEPLA) model, which is the first general framework to achieve deep integration of biochemical factual knowledge into the task of PLA prediction. The overall architecture of KEPLA follows an encoder-decoder paradigm. Specifically, it employs hybrid encoders—evolutionary sequence model (ESM) [19] for 1D amino acid sequences and graph convolutional networks (GCNs) [20] for 2D molecular graphs—to generate both global and local representations of proteins and ligands. To ensure the encoders effectively capture and leverage biochemical knowledge for the task, KEPLA jointly trains the encoded protein and ligand representations on two objectives: KG embedding and PLA prediction. For the KG embedding objective, the encoded global representations are directly used as the entity embeddings of our constructed KG, which covers protein-GO and ligand-LP knowledge. Then they are optimized alongside other learnable entity and relation embeddings using standard KG embedding techniques. For the PLA prediction objective, the encoded local representations are processed by a pairwise interaction module, which uses a cross attention network [21] to derive protein-ligand joint representations. The joint representations are then decoded by a multilayer perceptron (MLP) decoder to produce the final affinity prediction. By jointly minimizing the losses of both objectives, our model takes into account the relevant biochemical knowledge while making PLA predictions and thus can be enhanced. We conduct extensive experiments under both in-domain and cross-domain settings, comparing KEPLA with state-of-the-art methods. Results demonstrate that KEPLA achieves significant performance gains, improving RMSE metric by 5.28% and 12.42% on two benchmark datasets, respectively. Moreover, KEPLA provides more interpretable insights into the prediction through its knowledge-grounded framework.

To summarize, our contributions are as follows: (1) We propose KEPLA, the first knowledge-enhanced PLA predicting model that incorporates a KG with the interaction-free framework, achieving notable performance improvements. (2) By leveraging the KG and cross attention mechanism, KEPLA provides an interpretable prediction through relevant knowledge and attention visualization, moving beyond traditional black-box outputs. (3) We construct and release a novel KG dataset based on PDBbind, promoting the research on PLA prediction.

II. Related Work

A. Protein-Ligand Binding Affinity Prediction

Existing machine learning approaches for PLA prediction can be broadly categorized into two groups based on their core assumptions: interaction-free methods and interaction-based methods.

Interaction-free methods are primarily designed to simplify the modeling process by ignoring the intrinsic 3D structural and interaction information within protein-ligand complexes [22]. For instance, DeepDTA [9] directly employs raw ligand and protein sequences as inputs and utilizes a dual-channel CNN for feature extraction. GraphDTA [23] and MGraphDTA [12] represent ligands as 2D molecular graphs to preserve topological information, while DGraphDTA [11] enhances 1D protein sequence representations through 2D distance maps or protein graphs. Although these models offer computational efficiency by omitting 3D and atomic-level interactions, their generalization ability and interpretability remain limited [24].

In contrast, interaction-based methods explicitly incorporate 3D structural and atomic interaction information as prior knowledge to improve model robustness and interpretability. Early approaches, such as OnionNet [25], typically employ atom-atom pair features for model training, while mathematical tools like topological data analysis have also been introduced for molecular characterization [26], [27]. More recent techniques based on GNNs or 3D-CNNs simplify input representations by using fundamental atomic features such as element types and pairwise distances. For example, Pafnucy [28] adopts 3D-CNNs to extract spatial features from voxelized 3D grids, though their efficiency is hindered by sparse voxel occupancy. Recently, interaction graph neural network (IGNN) models, including SIGN [29] and GIANT [30], have attracted attention for their relational reasoning capabilities, learning

intra- and inter-molecular interactions in a staged or type-specific manner. However, these models remain computationally expensive and heavily reliant on accurate structural data, thus limiting their performance in affinity prediction tasks involving novel targets or incomplete interaction information.

B. Knowledge Graph

KG is a structured semantic network that reveals relations between entities, used to enhance machine cognition and intelligent applications [18]. In recent years, KGs have been gradually introduced into AI4S fields such as protein and molecular research as an effective means of integrating multi-source biological information. By organizing heterogeneous entities and relations—such as proteins, small molecule targets, biological pathways, and experimental data—into structured graphs, KGs provide rich background knowledge and semantic constraints for AI4S tasks [31].

In the field of PLA prediction, existing methods have attempted to jointly model prior knowledge (such as protein families, functional sites, molecular substructures, and pathway relations) with sequence and structural features [32]. By leveraging KG embeddings or GNNs, these methods incorporate global biological context into molecular representations, thereby improving prediction performance compared to relying solely on sequences or structures. Such approaches typically combine KG representations with molecular fingerprints, protein sequence embeddings, or 3D conformational features through concatenation or attention-based fusion, finally obtaining holistic representations that integrate both structural and biological information.

However, existing methods still face limitations: the integration of symbolic knowledge from KGs with physicochemical-based structural features remains nascent, and their robustness is inadequate for scenarios with scarce samples or novel targets. To address these issues, this work proposes an efficient KG-enhanced framework that tightly integrates semantic knowledge with structural information to improve PLA prediction performance.

III. Methods

A. Problem Formulation and Architecture

In PLA prediction, the goal is to estimate the binding strength between a target protein and a ligand compound. Each protein sequence is represented as $\mathcal{P} = (a_1, a_2, ..., a_K)$, where a_i denotes one of the 23 amino acids. For ligand compounds, although most existing deep learning methods encode molecular information using SMILES strings [8], their linear format does not explicitly capture molecular structure and may lead to the loss of important spatial information. To address this, we convert each SMILES string into a 2D molecular graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V}$ represents atoms and $\mathcal{E}$ denotes chemical bonds. Given a protein sequence $\mathcal{P}$ and a ligand graph $\mathcal{G}$, the PLA prediction task aims to learn a model that maps $\mathcal{P} \times \mathcal{G}$ to a continuous binding affinity value $y \in \mathbb{R}$.

Based on this formulation, the proposed KEPLA framework is illustrated in Figure 1. KEPLA consists of four main components: (1) ESM-based protein sequence encoding for learning protein representations; (2) GCN-based molecular graph encoding for learning ligand representations; (3) KG embedding objective, enhancing representations through external biochemical KG; (4) PLA prediction objective, enabling fine-grained modeling of protein-ligand interactions via cross attention. Subsequently, we present a detailed elucidation of these components.

B. Protein Sequence Encoding

The protein encoder is built upon the pre-trained ESM model [19], augmented with a multi-layer neural network. It transforms an input amino acid sequence into a set of latent vectors, each representing a local fragment of the protein. ESM, a widely recognized protein language model, has demonstrated strong performance in predicting protein structure, function, and other properties from individual sequences. To fully leverage ESM's powerful encoding capabilities, we use it as the backbone of our protein encoder. Given a target protein sequence $\mathcal{P}$, we input it into the pre-trained ESM model and extract the final hidden state as the initial feature matrix:

$$\mathbf{M}_p = \text{ESM}(\mathcal{P}), \tag{1}$$

where $\mathbf{M}_p \in \mathbb{R}^{D_p \times K}$, with D_p denoting the output dimension of ESM and K representing the maximum allowed sequence length. To support batch training, protein sequences longer than K are truncated, while shorter ones are zero-padded.

Since the initial features are high-dimensional and task-agnostic, they require further processing. To compress the representation and retain local sequence information, we apply average pooling over non-overlapping fragments of size s, yielding s-mer residue-level features:

$$\mathbf{X}_p = \text{AvgPool}_s(\mathbf{M}_p), \tag{2}$$

where $\mathbf{X}_p \in \mathbb{R}^{D_p \times M}$, with $M = K/s$ denoting the number of local fragments. Next, a multi-layer deep neural network (DNN) is employed to adapt these features to our task and extract higher-level abstractions of local protein fragments. The encoding process is defined as:

$$\mathbf{H}_p^{(l+1)} = \sigma(\text{DNN}(\mathbf{W}_n^{(l)}, \mathbf{b}_n^{(l)}, \mathbf{H}_p^{(l)})), \tag{3}$$

where $\mathbf{W}_n^{(l)}$ and $\mathbf{b}_n^{(l)}$ are the learnable weights and biases of the l-th DNN layer, $\mathbf{H}_p^{(l)}$ is the l-th hidden protein representation, and $\sigma(\cdot)$ is a non-linear activation function (ReLU in our case). The initial input is set as $\mathbf{H}_p^{(0)} = \mathbf{X}_p$. The final output, $\mathbf{H}_p \in \mathbb{R}^{D \times M}$, is adopted as the protein representation, where D is the representation dimension.

C. Molecular Graph Encoding

For each ligand compound, we convert its SMILES string into a 2D molecular graph $\mathcal{G}$. To represent the node (atom) information in $\mathcal{G}$, we initialize each atom using its chemical properties, following the implementation in the DGL-LifeSci library [33]. Each atom is initially represented as a 74-dimensional integer vector describing eight pieces of information: atom type, degree, number of implicit hydrogens, formal charge, number of radical electrons, hybridization

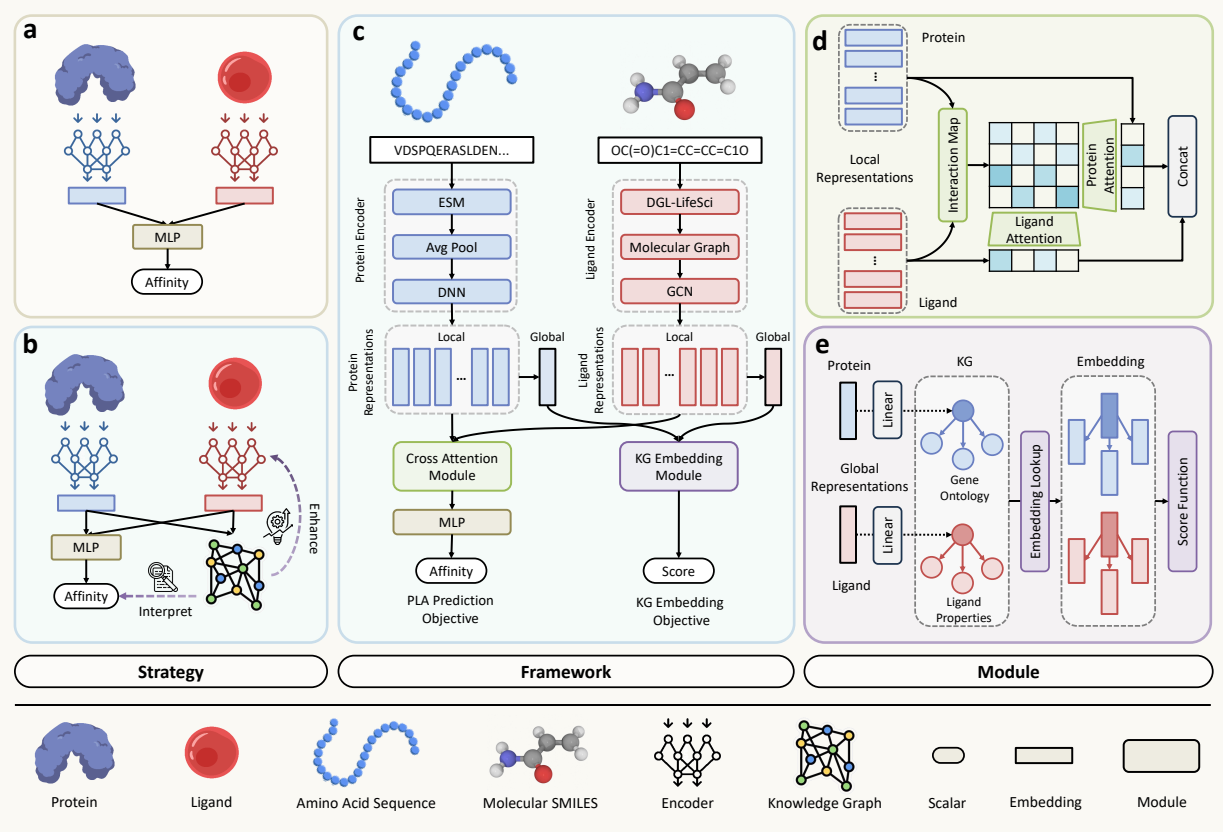


Fig. 1. **Overview of the KEPLA framework. (a)** Existing PLA prediction strategy. **(b)** KEPLA strategy, which incorporates a biochemical KG to enhance protein and ligand encoding. **(c)** KEPLA framework with two encoders: an ESM-based encoder for protein sequences and a GCN-based encoder for ligand SMILES, producing local and global representations. Global representations are used for KG embedding, while local representations are fed into the cross attention module for PLA prediction. **(d)** Cross attention module, where an interaction map derived from protein–ligand local representations enables fine-grained feature fusion, followed by an MLP decoder for affinity prediction. **(e)** KG embedding module, which projects global representations into a semantic space and evaluates triplet plausibility via a scoring function.

state, number of total hydrogen atoms, and aromaticity. This initialization is formally written as:

$$\mathbf{M}_d = \text{DGL-LifeSci}(\mathcal{G}), \tag{4}$$

resulting in the initial node feature matrix $\mathbf{M}_d \in \mathbb{R}^{74\times N}$, where N is the maximum allowed number of atoms per molecule. Molecules with fewer atoms are padded with virtual nodes filled with zeros to ensure consistent dimensions for batch training. Next, a linear transformation is applied to project the integer feature matrix into a real-valued one:

$$\mathbf{X}_d = \mathbf{W}\mathbf{M}_d, \tag{5}$$

producing a dense feature matrix $\mathbf{X}_d \in \mathbb{R}^{D_d\times N}$, with D_d denoting the feature dimension.

To learn expressive representations of ligand molecules, we employ a multi-layer GCN block. GCN extends convolution operations to irregular graph structures by updating each node's representation through aggregation of its neighboring nodes, effectively capturing molecular substructure information. We retain node-level representations to enable subsequent fine-grained modeling of interactions with protein fragments. The GCN-based ligand encoder is formulated as:

$$\mathbf{H}_d^{(l+1)} = \sigma(\text{GCN}(\mathbf{A}, \mathbf{W}_g^{(l)}, \mathbf{b}_g^{(l)}, \mathbf{H}_d^{(l)})), \tag{6}$$

where $\mathbf{W}_g^{(l)}$ and $\mathbf{b}_g^{(l)}$ are the learnable weight matrix and bias vector of the l-th GCN layer, $\mathbf{H}_d^{(l)}$ is the l-th hidden node representation, $\mathbf{A}$ is the adjacency matrix of the graph $\mathcal{G}$ (augmented with self-loops), and $\sigma(\cdot)$ denotes a non-linear activation function (ReLU in our case). The input to the first layer is set as $\mathbf{H}_d^{(0)} = \mathbf{X}_d$. The final output, $\mathbf{H}_d \in \mathbb{R}^{D\times N}$, is adopted as the ligand representation.

D. KG Embedding Objective

Thus far, the two encoders described above only capture the 1D and 2D structural information of proteins and ligands, respectively. To further enhance these encoders with biochemical factual knowledge, we integrate their outputs into a KG embedding framework. Specifically, we treat the encoded protein and ligand representations as entity embeddings within our constructed KG and incorporate them into a joint learning process alongside other entities and relations. The remainder of this subsection details our implementation.

Our KG structurally consists of *entity-relation-entity* triplets to describe relational knowledge. We define a triplet as (h, r, t), where h and t are head and tail entities, and r is the relation linking them, whose type is usually predefined by the schema. This KG is composed of two distinct subgraphs: one centered around proteins and the other around ligands.

- **Protein KG (GO-based)**: This subgraph incorporates two types of entities: protein entities $e_{protein}$ involved in the affinity prediction task, and Gene Ontology (GO) entities e_{GO} describing functional annotations. The GO entities cover three categories: Molecular Function (MF), Cellular Component (CC), and Biological Process (BP). Therefore, the protein-related triplets fall into three groups: *protein-MF*, *protein-CC*, and *protein-BP*.
- **Ligand KG (LP-based)**: Similarly, this subgraph includes ligand entities e_{ligand} and ligand property (LP) entities e_{LP}. The LP entities are further divided into Molecular Descriptors (MD), i.e., numerical attributes derived from molecular structures (e.g., counts of hydrogen bond donors/acceptors), and Chemical Features (CF), which denote qualitative attributes (e.g., hydrophobe, positively charged). Accordingly, the ligand-related triplets fall into two groups: *ligand-MD* and *ligand-CF*.

Next, we involve the protein and ligand representations in the KG embedding objective to enhance the protein and ligand encoding with external knowledge. Since the encoders yield matrix-form local representations unsuitable for KG embeddings, we first compute vector-form global representations by averaging across the local representations, then project them into the KG embedding space via a linear transformation:

$$\mathbf{h}_p = \mathbf{W}_k^1 \text{Average}(\mathbf{H}_p) + \mathbf{b}_k^1, \quad \mathbf{h}_d = \mathbf{W}_k^2 \text{Average}(\mathbf{H}_d) + \mathbf{b}_k^2, \tag{7}$$

where $\text{Average}(\cdot)$ denotes the averaging operation over all local representations. $\mathbf{W}_k^*$ and $\mathbf{b}_k^*$ are learnable weight matrices and bias vectors that map the global representations into the KG embedding space. The resulting $\mathbf{h}_p$ and $\mathbf{h}_d \in \mathbb{R}^D$ serve as the semantic embeddings of the protein and ligand entities in the KG. Similar to [34], the embedding sets of other entities and relations are denoted by learnable $\mathbf{T} \in \mathbb{R}^{D \times T}$ and $\mathbf{R} \in \mathbb{R}^{D \times R}$, where T is the number of non-protein and non-ligand entities, and R is the number of relation types.

To achieve the KG embedding objective, we train the KG by optimizing the corresponding embeddings to fit the triplet relations. The objective function is defined as:

$$\mathcal{L}_{KGE} = \frac{1}{|\mathcal{S}|} \sum_{(h,r,t) \in \mathcal{S}} F(h, r, t), \tag{8}$$

where $\mathcal{S}$ is the set of all KG triplets, and (h, r, t) is one of the triplets. $F(\cdot)$ is the score function of the KG embedding model. For modeling protein and ligand subgraphs, we adopt two widely used KG embedding models. RotatE [35] for the protein-GO subgraph:

$$F_p(h, r, t) = ||\mathbf{h} \circ \mathbf{r} - \mathbf{t}||, \tag{9}$$

where $\circ$ denotes the Hadamard (element-wise) product. TransE [34] for the ligand-LP subgraph:

$$F_d(h, r, t) = ||\mathbf{h} + \mathbf{r} - \mathbf{t}||. \tag{10}$$

In both functions, $\mathbf{h}$ is the embedding of the head entity, which is always a protein or ligand entity, and $\mathbf{t}$ is the embedding of the tail entity, which is always a GO or LP entity. $\mathbf{r}$ denotes the embedding of the relation between the head and the tail entities.

E. PLA Prediction Objective

In parallel with KG embedding, we employ a cross attention module to capture fine-grained, pairwise local interactions between protein and ligand substructures. This module comprises two main steps: (1) constructing an interaction map to measure substructure-level similarity, and (2) applying cross attention to derive a joint protein-ligand representation.

For the first step, ESM and GCN encoders have already generated hidden representations for protein and ligand: $\mathbf{H}_p = \{\mathbf{h}_p^1, \mathbf{h}_p^2, ..., \mathbf{h}_p^M\}$ and $\mathbf{H}_d = \{\mathbf{h}_d^1, \mathbf{h}_d^2, ..., \mathbf{h}_d^N\}$, where M and N are the number of fragments in a protein and the number of atoms in a ligand, respectively. Using these hidden representations, we construct a pairwise interaction matrix $\mathbf{V} \in \mathbb{R}^{M \times N}$:

$$\mathbf{V} = \mathbf{H}_p^\top \mathbf{H}_d. \tag{11}$$

Each element in $\mathbf{V}$ reflects the interaction strength between a specific protein fragment and a ligand atom, thus identifying potential binding regions.

To extract a joint representation informed by these local interactions, we treat the interaction matrix as a similarity matrix and compute cross attention weights over both protein and ligand substructures. Protein attention $\boldsymbol{\alpha}_p \in \mathbb{R}^M$ is calculated by summing the interactions across all ligand atoms and normalizing via a Softmax function:

$$\boldsymbol{\alpha}_p = \text{Softmax}(\frac{1}{\sqrt{D}} \cdot \sigma(\frac{1}{N} \cdot \text{Sum}(\mathbf{V}))), \tag{12}$$

where $\text{Sum}(\cdot)$ denotes column-wise summation in matrix $\mathbf{V}$, and the result accumulates each fragment's interactions across ligand atoms. $\sigma(\cdot)$ selects the Tanh function, and the $\text{Softmax}(\cdot)$ normalizes the attention scores into a probability distribution. The scaling factor $\sqrt{D}$ helps stabilize gradients. Ligand attention $\boldsymbol{\alpha}_d \in \mathbb{R}^N$ is computed analogously by transposing the interaction matrix and aggregating over protein fragments:

$$\boldsymbol{\alpha}_d = \text{Softmax}(\frac{1}{\sqrt{D}} \cdot \sigma(\frac{1}{M} \cdot \text{Sum}(\mathbf{V}^\top))). \tag{13}$$

These attention weights can provide interpretability by highlighting which protein and ligand substructures contribute most to the interaction.

Using the attention weights, we compute the weighted sum of protein and ligand local representations and concatenate them to form a joint representation:

$$\mathbf{f}_{p,d} = \mathbf{H}_p \cdot \boldsymbol{\alpha}_p \oplus \mathbf{H}_d \cdot \boldsymbol{\alpha}_d, \tag{14}$$

where $\mathbf{f}_{p,d} \in \mathbb{R}^{2D}$ is the joint protein-ligand representation, and $\oplus$ denotes vector concatenation. We feed this joint representation into an MLP decoder to predict the binding affinity:

$$\hat{y}_{p,d} = \text{MLP}(\mathbf{f}_{p,d}), \tag{15}$$

where each single layer consists of its own learnable weight matrix and bias vector, followed by a ReLU activation function. The PLA objective is defined as the mean absolute error between the predicted and true binding affinities:

$$\mathcal{L}_{PLA} = \frac{1}{|\mathcal{O}|} \sum_{(p,d) \in \mathcal{O}} |y_{p,d} - \hat{y}_{p,d}|, \tag{16}$$

where $\mathcal{O}$ is the set of protein-ligand training pairs, and $y_{p,d}$ is the ground truth affinity.

To jointly learn structural and semantic knowledge, we combine the affinity prediction objective with the KG embedding objective. The overall training loss function to minimize is as follows:

$$\mathcal{L} = \mathcal{L}_{PLA} + \beta\mathcal{L}_{KGE} + \lambda||\mathbf{\Theta}||_2^2, \tag{17}$$

where $\mathbf{\Theta}$ denotes the set of all learnable parameters, β is a trade-off hyperparameter between tasks, and λ is the L_2 regularization coefficient. We use mini-batch Adam optimization [36] to train the model. For each mini-batch of sampled pairs $(p, d) \in \mathcal{O}$, we select the associated KG triplets, construct their representations, and update all model parameters via backpropagation using the gradient of the total loss.

IV. Experimental Setup

A. Datasets

We evaluate KEPLA and baselines using the following publicly available benchmark datasets for PLA prediction. PDBbind is a widely used dataset that provides multi-dimensional binding structures of protein-ligand complexes along with experimentally determined binding affinities. In our experiments, we primarily use PDBbind v2016 [37], which is the most frequently adopted version in recent studies. This version consists of three overlapping subsets: The *general set*, containing all 13,283 protein-ligand complexes; The *refined set*, comprising 4,057 higher-quality complexes selected from the general set; The *core set*, including 290 top-quality complexes curated through a rigorous selection process, serving as a benchmark for testing. For convenience, we refer to the 3,767 complexes that belong to the *refined set* but are not in the *core set* as the *refined set* in this paper. To further assess generalization ability, we additionally evaluate on the CSAR-HiQ [38] dataset, an external benchmark consisting of two subsets with 176 and 167 protein-ligand complexes, respectively, sourced independently from PDBbind.

Table I summarizes the entity and triplet types in our constructed KG and their statistics. The KG mainly encodes Gene Ontology (GO) knowledge for proteins and ligand property (LP) knowledge for ligands, and consists of triplets involving all proteins and ligands in the *refined set* of PDBbind dataset. Protein-related triplets are constructed by retrieving GO terms from the Biological Process, Cellular Component, and Molecular Function categories via UniProt[1], using protein UniProt IDs, with the top 1,000 most frequent terms retained as GO entities. Ligand-related triplets are derived from RDKit-parsed information based on ligand MOL files, including molecular descriptors (e.g., hydrogen bond donors/acceptors, ring structures, rotatable bonds, stereochemistry, and heteroatoms) and chemical feature types, which are treated as LP entities.

TABLE I
STATISTICS OF ENTITIES AND TRIPLETS IN THE KNOWLEDGE GRAPH.

Entity types	Count	Triplet types	Count
Protein	2,542	Protein–GO	11,859
Biological Process	580	Protein–Biological Process	4,827
Cellular Component	165	Protein–Cellular Component	3,598
Molecular Function	255	Protein–Molecular Function	3,434
Ligand	3,181	Ligand–LP	48,356
Molecular Descriptor	175	Ligand–Molecular Descriptor	28,629
Chemical Feature	23	Ligand–Chemical Feature	19,727

B. Baselines

We compare KEPLA with a wide range of baseline methods. All baselines are implemented using official codebases and evaluated on the same dataset splits.

[1]https://www.uniprot.org

The first group consists of *interaction-free* baselines, which predict binding affinity without access to explicit protein-ligand interactions.

- **RF-Score** [22] is a classical machine learning method that uses handcrafted intermolecular features and a random forest for affinity prediction.
- **DeepDTA** [9] employs CNNs to extract features from protein sequences and ligand SMILES strings, without relying on structural information.
- **GraphDTA** [23] models ligands as molecular graphs and proteins as sequences. It includes four variants based on different GNN backbones: GCN, GAT, GIN, and GAT-GCN.
- **DrugBAN** [24] encodes drugs and proteins using GCN and CNN, respectively, and applies bilinear attention to model their interactions without explicit 3D structure modeling.

The second group includes *interaction-based* baselines, which explicitly leverage available protein-ligand interaction information during modeling.

- **Pafnucy** [28] is a 3D CNN-based model that learns spatial features from voxelized protein–ligand complexes.
- **OnionNet** [25] extracts element-pair contact features organized in concentric shells and applies CNNs for affinity prediction.
- **SGCN** [39] incorporates atomic coordinates into graph convolutions to explicitly model 3D molecular structures.
- **GraphTrans** [40] combines GNNs with Transformer-based self-attention to capture global topological dependencies.
- **NL-GCN** [41] introduces non-local message passing with attention-based aggregation for global structure modeling.
- **GNN-DTI** [26] is a geometry-aware graph attention model that incorporates interatomic distance information.
- **DMPNN** [27] performs message passing over directed bonds, emphasizing edge-level information.
- **MAT** [42] adopts a Transformer architecture augmented with interatomic distances for molecular representation learning.
- **DimeNet** [43] explicitly encodes angular and radial information for precise geometric modeling.
- **CMPNN** [44] enhances DMPNN by enabling bidirectional communication between nodes and edges.
- **SIGN** [29] is a simplified geometric interaction network that incorporates distances and angles for affinity prediction.
- **GIANT** [30] is a state-of-the-art PLA model that jointly models 3D geometric graphs and interaction networks.

TABLE II
IN-DOMAIN PERFORMANCE COMPARISON ON THE PDBBIND AND CSAR-HIQ DATASETS UNDER RANDOM SPLIT (STATISTICS OVER FIVE INDEPENDENT RUNS)

Method		PDBbind core set				CSAR-HiQ set			
		RMSE ↓	MAE ↓	SD ↓	R ↑	RMSE ↓	MAE ↓	SD ↓	R ↑
Interaction-free methods	RF-Score	1.446 (0.008)	1.161 (0.007)	1.335 (0.010)	0.789 (0.003)	1.947 (0.012)	1.466 (0.009)	1.796 (0.020)	0.723 (0.007)
	DeepDTA	1.639 (0.026)	1.315 (0.023)	1.689 (0.025)	0.718 (0.014)	1.983 (0.036)	1.510 (0.059)	2.037 (0.051)	0.633 (0.036)
	GCN	1.735 (0.034)	1.343 (0.037)	1.719 (0.027)	0.613 (0.016)	2.324 (0.079)	1.732 (0.065)	2.302 (0.061)	0.464 (0.047)
	GAT	1.765 (0.026)	1.354 (0.033)	1.740 (0.027)	0.601 (0.016)	2.213 (0.053)	1.651 (0.061)	2.215 (0.050)	0.524 (0.032)
	GIN	1.640 (0.044)	1.261 (0.044)	1.621 (0.036)	0.667 (0.018)	2.158 (0.074)	1.624 (0.058)	2.156 (0.088)	0.558 (0.047)
	GAT-GCN	1.562 (0.022)	1.191 (0.016)	1.558 (0.018)	0.697 (0.008)	1.980 (0.055)	1.493 (0.046)	1.969 (0.057)	0.653 (0.026)
	DrugBAN	1.378 (0.025)	1.102 (0.019)	1.328 (0.020)	0.788 (0.015)	1.672 (0.034)	1.334 (0.042)	1.728 (0.059)	0.767 (0.039)
Interaction-based methods	Pafnucy	1.585 (0.013)	1.284 (0.021)	1.563 (0.022)	0.695 (0.011)	1.939 (0.103)	1.562 (0.094)	1.885 (0.071)	0.686 (0.027)
	OnionNet	1.407 (0.034)	1.078 (0.028)	1.391 (0.038)	0.768 (0.014)	1.927 (0.071)	1.471 (0.031)	1.877 (0.097)	0.690 (0.040)
	SGCN	1.583 (0.033)	1.250 (0.036)	1.582 (0.320)	0.686 (0.015)	1.902 (0.063)	1.472 (0.067)	1.891 (0.077)	0.686 (0.030)
	GraphTrans	1.539 (0.044)	1.182 (0.046)	1.521 (0.042)	0.714 (0.019)	1.950 (0.072)	1.508 (0.069)	1.886 (0.083)	0.687 (0.033)
	NL-GCN	1.516 (0.019)	1.198 (0.013)	1.511 (0.024)	0.720 (0.010)	1.840 (0.024)	1.393 (0.016)	1.817 (0.028)	0.716 (0.011)
	GNN-DTI	1.492 (0.025)	1.192 (0.032)	1.471 (0.051)	0.736 (0.021)	1.972 (0.061)	1.547 (0.058)	1.834 (0.090)	0.709 (0.035)
	DMPNN	1.493 (0.016)	1.188 (0.009)	1.489 (0.014)	0.729 (0.006)	1.886 (0.026)	1.488 (0.054)	1.865 (0.035)	0.697 (0.013)
	MAT	1.457 (0.037)	1.154 (0.037)	1.445 (0.033)	0.747 (0.013)	1.879 (0.065)	1.435 (0.058)	1.816 (0.083)	0.715 (0.030)
	DimeNet	1.453 (0.027)	1.138 (0.026)	1.434 (0.023)	0.752 (0.010)	1.805 (0.036)	1.338 (0.026)	1.798 (0.027)	0.723 (0.010)
	CMPNN	1.408 (0.028)	1.117 (0.031)	1.399 (0.025)	0.765 (0.009)	1.839 (0.096)	1.411 (0.064)	1.767 (0.103)	0.730 (0.052)
	SIGN	1.316 (0.031)	1.027 (0.025)	1.312 (0.035)	0.797 (0.012)	1.735 (0.031)	1.327 (0.040)	1.709 (0.044)	0.754 (0.014)
	GIANT	1.269 (0.020)	0.999 (0.018)	1.265 (0.024)	0.814 (0.008)	1.666 (0.024)	1.242 (0.030)	1.633 (0.034)	0.779 (0.011)
Ours	KEPLA	**1.202 (0.017)**	**0.936 (0.013)**	**1.195 (0.016)**	**0.832 (0.005)**	**1.459 (0.019)**	**1.132 (0.026)**	**1.517 (0.032)**	**0.813 (0.012)**

Results are reported as mean (standard deviation). For each dataset and metric, the best performance is highlighted in bold, and the second-best is underlined.

C. Evaluation Strategies and Metrics

We evaluate prediction performance on the two public datasets using two distinct data split strategies to simulate in-domain and cross-domain scenarios. For in-domain evaluation, following [22], we use the *refined set* of PDBbind as the primary training data, as there is substantial overlap between the full *general set* and CSAR-HiQ dataset. We randomly split the protein-ligand complexes in the *refined set* into training and validation sets at a 9:1 ratio. For testing, we use the PDBbind *core set* and the CSAR-HiQ dataset after removing any overlapping complexes from the *refined set*.

For cross-domain evaluation, we adopt a clustering-based pair split strategy on the large-scale PDBbind dataset to simulate realistic domain shift. Proteins and ligands are clustered independently using single-linkage hierarchical clustering, ensuring a minimum inter-cluster distance γ to avoid overly similar samples. Proteins are represented by PSC (pseudo-amino acid composition), while ligands are encoded using ECFP4 (extended connectivity fingerprint, up to four bonds). Cosine distance and Jaccard distance are used for proteins and ligands, with thresholds $\gamma = 0.001$ and $\gamma = 0.5$, respectively. We randomly select 60% of protein clusters and 60% of ligand clusters to form the source domain, and use the remaining clusters to construct the target domain. The source and target domains are disjoint and follow different distributions. For training and validation, we use all source data and 80% of the target data, while the remaining 20% of the target data is reserved for testing. This cross-domain setting is more challenging than random splits and better reflects real-world generalization in drug discovery.

To comprehensively evaluate the model performance, following [25], [28], we adopt Root Mean Square Error (RMSE), Mean Absolute Error (MAE), Pearson's correlation coefficient (R), and the standard deviation (SD) in regression to measure the prediction error. For evaluation, lower RMSE, MAE, and SD values indicate better performance, whereas higher R values are preferable.

D. Implementation

KEPLA is implemented using Python 3.10 and PyTorch 2.1.2, along with functionalities from several auxiliary libraries, including DGL 2.4.0[2], DGL-LifeSci 0.3.2[3], and RDKit 2024.03.5[4]. The protein encoder includes a pre-trained ESM2_t36_3B model, followed by two fully connected layers with dimensions [2560, 512, 128]. The average pooling window size is set to 9. The ligand encoder consists of three GCN layers with hidden dimensions of [128, 128, 128]. We cap the maximum allowed protein sequence length at 1,080 and the maximum number of ligand atoms at 290. In the cross attention module, we use a single attention head for simplicity. The latent embedding size is set to 128, and the fully connected decoder contains 512 hidden units. The trade-off value in the loss function is set to 0.1. We set the batch size to 64 and optimize the model using the Adam optimizer with a learning rate of 1e-4. All models are trained for a maximum of 200 epochs, and the checkpoint with the lowest validation RMSE is selected for final evaluation on the test set.

[2]https://www.dgl.ai
[3]https://github.com/awslabs/dgl-lifesci
[4]https://www.rdkit.org

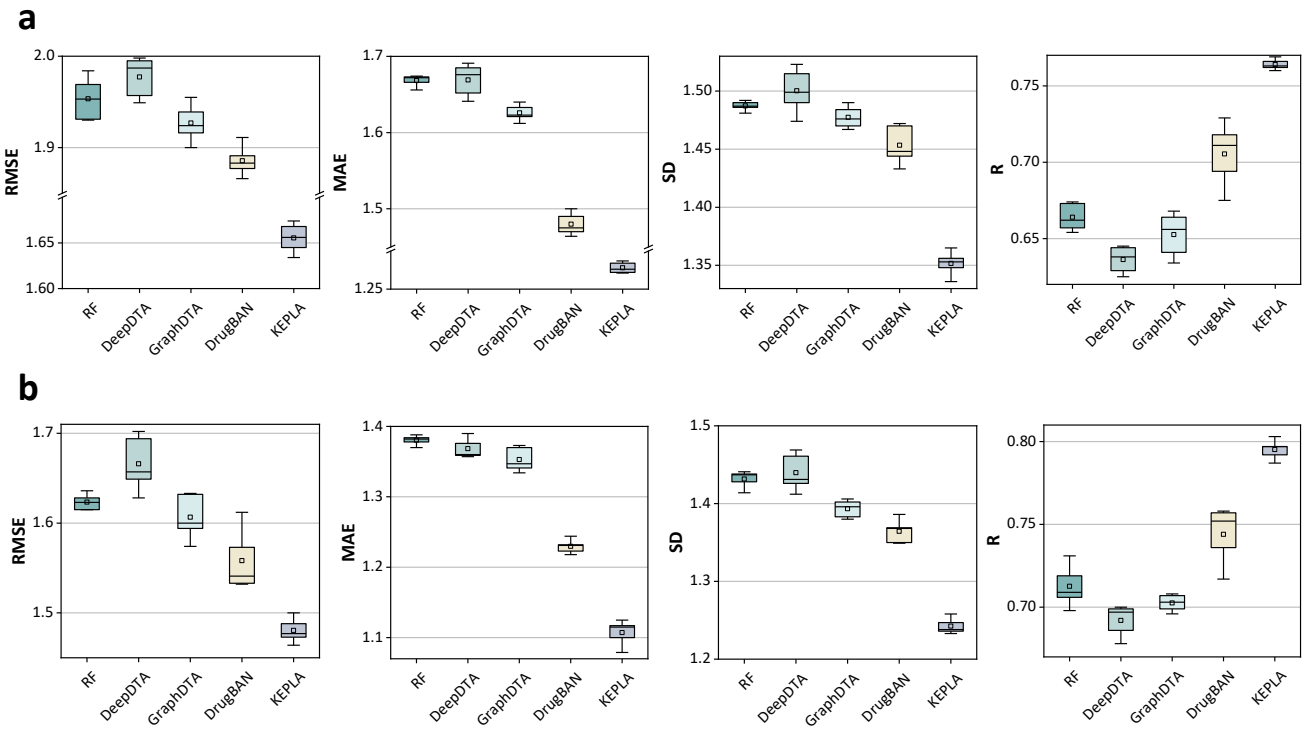


Fig. 2. **Cross-domain performance comparison on the PDBbind dataset (statistics over five independent runs). (a)** Performance comparison of interaction-free methods under clustering-based pair split. **(b)** Performance comparison of interaction-free methods under cold pair split. The box plots display the median as the center line and the mean as a square marker. The minimum and lower percentile indicate the worst and second-worst scores, while the maximum and upper percentile represent the best and second-best scores, respectively.

V. Experimental Results

A. In-Domain Performance Comparison

Table II summarizes the in-domain performance comparison under the random split setting. We report the mean and standard deviation of four evaluation metrics across five independent runs. When comparing model categories, we observe that interaction-based models generally outperform interaction-free models, highlighting the advantage of incorporating 3D structural information and physical interactions in PLA prediction. Despite being an interaction-free method, our KEPLA achieves the best performance across all experimental settings on both datasets, surpassing even all interaction-based baselines. Specifically, KEPLA improves RMSE by 5.28% on the PDBbind dataset and by 12.42% on the CSAR-HiQ dataset compared to the best-performing baseline GIANT. These results strongly support KEPLA's superior predictive capability. Considering KEPLA requires only the molecular graph of ligand compounds and primary sequences of target proteins as input like most interaction-free methods, the outstanding performance benefits from knowledge-enhanced representation learning and local interaction modeling.

B. Cross-Domain Performance Comparison

In-domain prediction under a random split is relatively easy and of limited practical relevance. We therefore focus on the more realistic and challenging task of cross-domain PLA prediction, where training and test data follow different distributions. To this end, we partition the PDBbind dataset into source and target domains using a clustering-based pair split strategy. As a complementary evaluation, we also adopt a cold pair split, in which 70% of proteins and ligands are randomly selected to form the training set, and the remaining 30% are split into validation and test sets with a 3:7 ratio. This setting ensures that neither proteins nor ligands in the test set are observed during training, providing a stringent evaluation of generalization.

The performance evaluation on the PDBbind dataset under both clustering-based and cold pair split strategies is shown in Figure 2. It is important to note that these experiments are restricted to interaction-free methods, as structural interaction information is unavailable in the simulated cross-domain and cold scenarios. Compared to the earlier in-domain results, the performance of all interaction-free PLA models drops significantly as shown in Figure 2a. This decline is primarily attributed to the reduced information overlap between the training and test sets, as well as the absence of sufficiently similar proteins and ligands to serve as references. Moreover, Figure 2b highlights a noticeable performance drop when moving from random split to cold pair split, emphasizing that prediction accuracy on unseen test data cannot solely rely on previously encountered protein or ligand features.

Despite the increased difficulty of cross-domain and cold settings, KEPLA consistently outperforms all state-of-the-art deep learning baselines. Specifically, it achieves improvements of 12.25% in RMSE and 8.37% in R over DrugBAN on the PDBbind dataset. These results highlight the robustness of KEPLA and underscore its strong generalization capability across

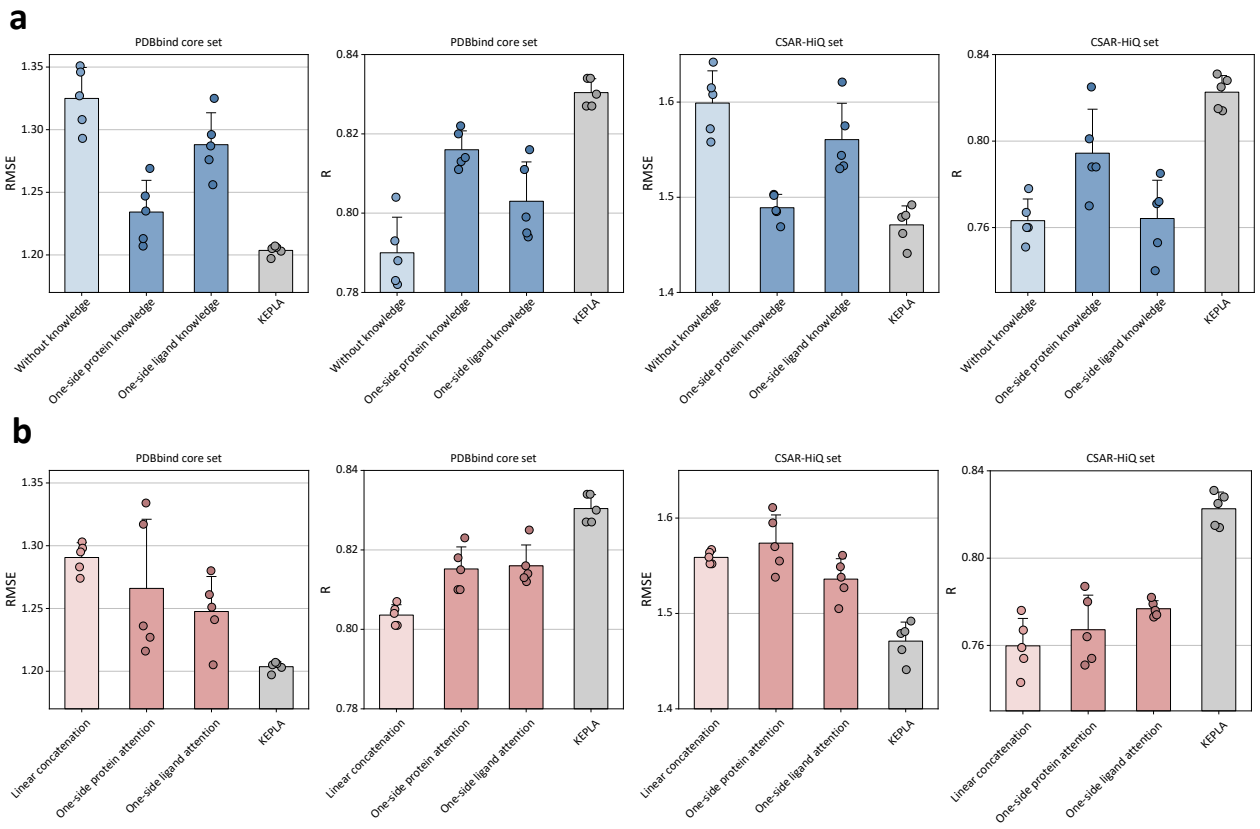


Fig. 3. **Ablation study in terms of RMSE and R on the PDBbind and CSAR-HiQ datasets under random split (statistics over five independent runs). (a)** Performance of different variants of the KG module. **(b)** Performance of different variants of the cross attention module. The vertical bars represent the mean values, while the black lines denote standard deviations (error bars). Dots indicate the performance scores from each individual run.

domains. Interestingly, the traditional RF-Score model also performs well, even consistently outperforming the deep learning baseline DeepDTA under cross-domain and cold settings. This observation suggests that deep learning approaches are not inherently superior to shallow machine learning models in challenging generalization scenarios. One possible explanation is that data-driven deep learning models are more sensitive to data bias and distribution shifts. Furthermore, we observe that KEPLA experiences a 36.55% performance drop compared to its random split performance, while DrugBAN shows a slightly higher average decline of 36.87%. This indicates that KEPLA's knowledge-enhanced design contributes to greater robustness, and supports the notion that integrating external knowledge can effectively mitigate the impact of data bias in deep learning-based PLA prediction.

C. Ablation Study

We conduct an ablation study to investigate the contributions of two key components in KEPLA: the KG and the cross attention modules. The results of various KEPLA variants are presented in Figure 3. To evaluate the impact of KG enhancement, we consider three variants of KEPLA with different configurations: KEPLA using only the protein-side KG, KEPLA using only the ligand-side KG, and KEPLA without KG enhancement. As shown in Figure 3a, the model's strong performance is clearly supported by the incorporation of external knowledge, validating the effectiveness of KG enhancement. Notably, the protein-side KG contributes more significantly to affinity prediction than the ligand-side counterpart. A likely explanation is that the protein KG contains more entities and richer relational information, offering more comprehensive context for representation learning.

To further assess the effectiveness of the cross attention mechanism, we evaluate three KEPLA variants that differ in how they compute joint representations between proteins and ligands: one-side protein attention, one-side ligand attention, and linear concatenation. In these variants, the cross attention module in KEPLA is replaced with either ligand-side or protein-side attention. The one-side attention variants adopt the neural attention mechanism proposed in [45], where the joint representation is derived from the interaction between a ligand vector and a protein fragment matrix. The linear concatenation variant, on the other hand, simply concatenates the protein and ligand vector representations following a max-pooling operation. As shown in Figure 3b, the results confirm that the cross attention mechanism is both effective and essential, as it enables the model to capture fine-grained interactions crucial for accurate PLA prediction. Among the one-side attention variants, ligand-side attention yields better performance. This may be attributed to the fact that ligand-side attention operates on local representations derived from the ligand's 2D molecular structure, which tend to be more spatially precise compared to the 1D sequence-based representations of proteins.

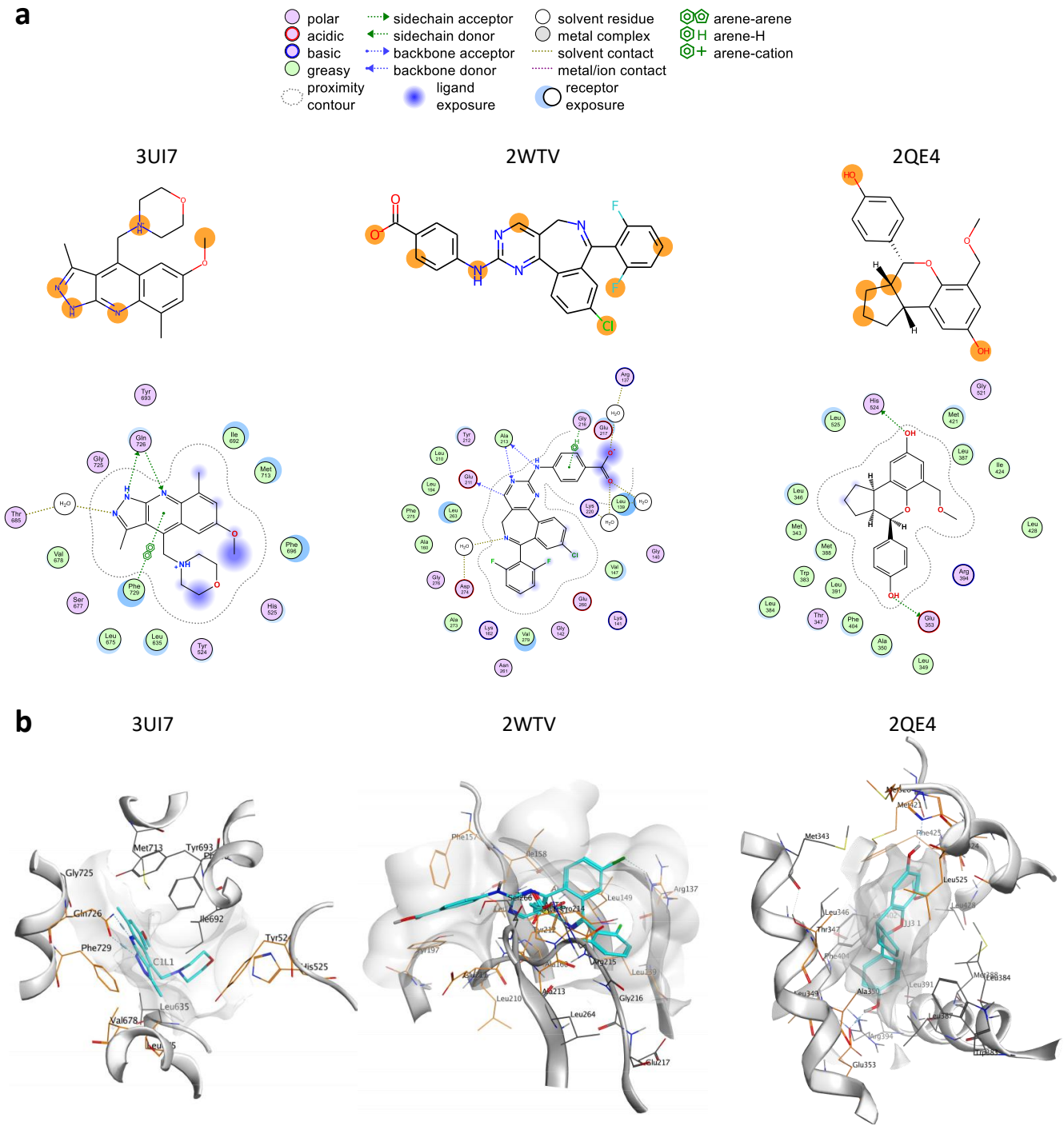


Fig. 4. **Visualization of ligands and binding pockets for structural-level interpretability study. (a)** Interpretability of co-crystallized ligands. The upper section of each panel displays the 2D structures of ligands, with atoms highlighted in orange to indicate those predicted to contribute to protein binding. All 2D structures are visualized using RDKit. The lower section of each panel presents protein-ligand interaction maps derived from the corresponding crystal structures. **(b)** Interpretability of binding pocket structures. The 3D representations show protein-ligand binding pockets, with correctly predicted binding-site residues highlighted in orange and the corresponding ligands shown in cyan. Remaining amino acid residues, secondary structure elements, and surface maps are shown in grey. All protein-ligand interaction maps and 3D visualizations of X-ray crystal structures are generated using the Molecular Operating Environment (MOE) software.

D. Structural-Level Interpretability with Cross Attention

A further strength of KEPLA lies in its ability to provide interpretability at both structural and knowledge levels—an essential feature for drug design applications. Structural-level interpretability is achieved by leveraging cross attention weights to visualize the contribution of each substructure to the final prediction. To illustrate this, we examine three top-ranked predictions (PDB IDs: 3UI7, 2WTV, and 2QE4) corresponding to co-crystallized ligands from the Protein Data Bank (PDB) [46]. Only X-ray structures with a resolution greater than $2.0\mathring{A}$ and targeting human proteins are considered [47]. Additionally, the selected complexes must exhibit binding affinities at the nanoscale and belong to the test set. Visualization results are presented in Figure 4a, alongside protein-ligand interaction maps derived from the corresponding X-ray structures. For each ligand, the top 20% of atoms, ranked by cross attention weights, are highlighted in orange.

For complex 3UI7 (pyrazoloquinoline bound to human PDE10A), KEPLA correctly identifies nitrogen atoms within the heterocyclic core as key hydrogen-bond donors and acceptors, matching experimental interactions with Gln729 and Thr685. The methoxy group, although solvent-exposed, is mistakenly highlighted but may contribute weakly to solvation-mediated interactions. The pyridine ring involved in an arene-arene contact with Phe729 is also partly emphasized. Notably, most of the top-ranked atoms correspond to verified interaction sites. For complex 2WTV (MLN8054 bound to human Aurora kinase A), the model highlights major binding motifs, including the aniline amino group (hydrogen bond donor to Ala213), carboxylate (interacting with Tyr212), and nitrogen-containing heterocycles (hydrogen bond to Glu211). The fluorine atoms in the difluorophenyl group and the chlorine substituent are also marked, likely due to solvent exposure and affinity effects. Despite the ligand's higher complexity, KEPLA accurately captures most key interactions observed in the crystal structure. For complex 2QE4 (benzopyran agonist complexed with the human estrogen receptor), KEPLA pinpoints hydroxyl groups forming hydrogen bonds with Glu353 and His524 and identifies several solvent-exposed carbons potentially involved in secondary interactions.

Protein-level interpretability is coarser, as attention is computed over residue fragments rather than single residues. Even so, the model successfully highlights key binding residues: Gln726 and Phe729 in 3UI7, Glu211 and Ala213 in 2WTV, and Glu353 and His524 in 2QE4, along with other pocket-forming residues shown in Figure 4b. Across complexes, nearly half of the experimentally verified binding-site residues are detected, suggesting fragment-level attention provides a solid basis for interpretability. Future incorporation of residue-level attention and local 3D structural cues may further improve resolution and explanatory power.

In addition, since the interpretability provided by KEPLA is adaptively learned from the PLA data itself, it has the potential to uncover previously unexplored local interaction patterns. Such insights could assist drug developers in optimizing the binding properties of a given scaffold or in minimizing the off-target liabilities of a compound.

E. Knowledge-Level Interpretability with Knowledge Graph

We further examine whether external knowledge can enhance the interpretability of protein-ligand binding. Specifically, we identify entities most closely related to the target protein or ligand by applying the KG scoring function to all associated triplets. Lower scores indicate stronger semantic relevance. As illustrated in Figure 5, the top five nearest GO or LP entities and their definitions are used to explain the binding mechanism. Consistent with prior experiments, we analyze the same case studies (PDB IDs: 3UI7, 2WTV, and 2QE4), discussing 3UI7 in detail and the others briefly due to space constraints.

For 3UI7 (pyrazoloquinoline bound to human PDE10A), the top GO entities most closely associated with the protein include *cAMP binding*, *cytosol*, *3',5'-cyclic-GMP phosphodiesterase activity*, *metal ion binding*, and *cGMP binding*. Definitions for these entities are provided in Figure 5a. Collectively, they suggest that PDE10A functions by catalyzing the hydrolysis of cAMP and cGMP in the cytosol, playing a role in intracellular signal transduction. This enzymatic activity is metal ion-dependent, specifically requiring coordination with Zn^{2+} and Mg^{2+} ions. The binding of the ligand pyrazoloquinoline to PDE10A is closely aligned with these biological functions. The ligand binds at the enzyme's catalytic domain, acting as a competitive inhibitor that blocks access to the natural substrates cAMP and cGMP, thereby preventing their hydrolysis. According to structural data from the PDB, Zn^{2+} and Mg^{2+} coordination sites in the catalytic pocket also participate in the protein-ligand interaction. On the ligand side, the top associated entities for pyrazoloquinoline are *two aromatic heterocycles*, *three rotatable bonds*, *positively charged nitrogen*, *four hydrogen bond acceptors*, and *one aromatic carbocycle*, with definitions provided in Figure 5b. These descriptors align well with the interpretability analysis shown in Figure 4a. Specifically, the presence of aromatic heterocycles suggests potential for π-π stacking or other aromatic interactions with residues such as Phe, along with hydrogen bonding and structural rigidity, factors known to positively influence binding affinity. Additionally, the ligand's rotatable bonds may contribute to conformational flexibility, enhancing its ability to adapt to the shape of the protein's binding pocket. Together, these findings help explain the strong binding affinity observed in complex 3UI7.

For 2WTV (MLN8054 bound to human Aurora kinase A), the top GO entities most closely associated with the protein include *protein kinase binding*, *cytosol*, *protein serine kinase activity*, *ATP binding*, and *protein phosphorylation*. These annotations show that Aurora kinase A is essential for spindle assembly and chromosome segregation during mitosis through its ATP-binding and serine/threonine kinase activities. The interaction between MLN8054 and Aurora kinase A aligns with these annotations. MLN8054 is a selective inhibitor that mimics ATP, binds competitively to its ATP site, blocking kinase activity and disrupting cell division. On the ligand side, the LP entities most closely associated with MLN8054 include *acidic group*, *three aromatic carbocycles*, *four rotatable bonds*, *nine heteroatoms*, and *one aromatic heterocycle*. They

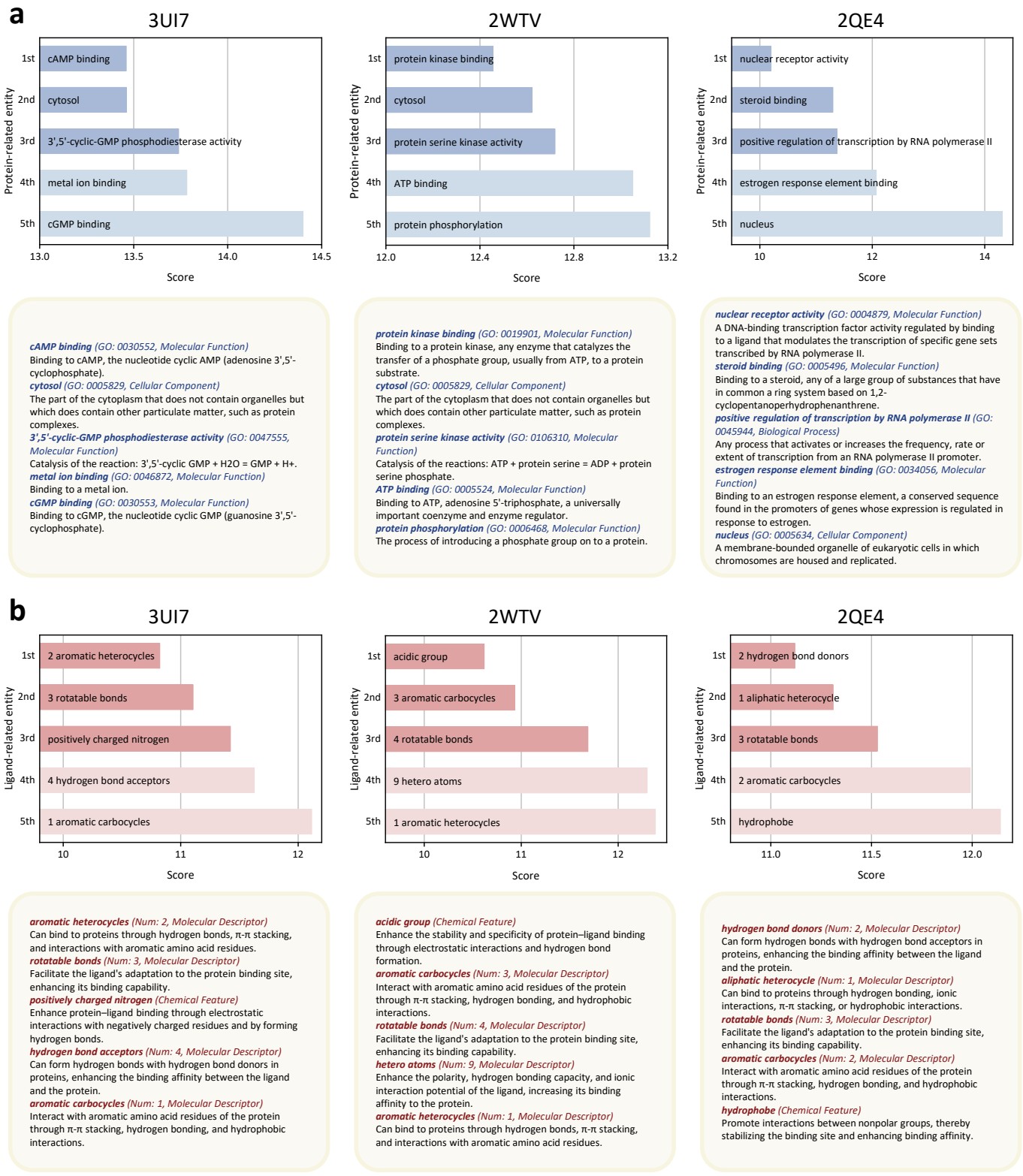


Fig. 5. **Biochemical knowledge of proteins and ligands for knowledge-level interpretability study. (a)** Interpretability of protein biological knowledge. The upper part of each panel displays the GO entities most closely associated with the protein, ranked in ascending order according to their triplet scores from the KG embedding model. The lower part presents the types and detailed definitions of these entities, annotated with their corresponding GO IDs. **(b)** Interpretability of ligand chemical knowledge. The upper part of each panel shows the LP entities most closely related to the ligand, also ranked in ascending order by their triplet scores in the KG embedding model. The lower part provides the types and specific functions of these entities, with molecular descriptors annotated by their corresponding numbers.

facilitate hydrogen bonding, metal coordination, and conformational flexibility, consistent with high binding affinity.

For 2QE4 (benzopyran agonist complexed with the human estrogen receptor), the top GO entities most closely associated with the protein are *nuclear receptor activity*, *steroid binding*, *positive regulation of transcription by RNA polymerase II*, *estrogen response element binding*, and *nucleus*. These annotations reflect the receptor's steroid hormone function. On the ligand side, the most relevant LP entities for the benzopyran agonist include *two hydrogen bond donors*, *one aliphatic heterocycle*, *three rotatable bonds*, *two aromatic carbocycles*, and *hydrophobe*. These features enhance hydrogen bonding and hydrophobic interactions that stabilize the complex.

The above examples show that KEPLA can effectively retrieve protein- and ligand-associated knowledge entities from the KG, enabling biologically and chemically grounded explanations of binding interactions. Compared to purely data-driven black-box predictions, this knowledge-based interpretability provides a more tangible and trustworthy foundation for rational drug discovery.

F. Generalization Ability Comparison

As PLA data continues to grow, the proportion of high-quality data in the *refined set* remains relatively limited. Consequently, a model's ability to effectively utilize a larger volume of lower-quality data becomes a critical measure of its generalization capability—an essential aspect of performance evaluation. To assess this, we conduct additional generalization experiments on the PDBbind *general set*, which contains all 13,283 protein-ligand complexes. As shown in Figure 6, KEPLA consistently achieves the lowest prediction errors under both training settings compared to mainstream baseline models. Notably, when trained on the *general set*, KEPLA improves RMSE and MAE by approximately 6.16% and 5.98%, respectively, further widening the performance gap with baseline models. These results highlight KEPLA's strong generalization ability to large-scale, lower-quality data.

VI. Conclusion

In this work, we present KEPLA, an end-to-end knowledge-enhanced deep learning framework for PLA prediction, designed to address the performance bottlenecks and limited interpretability of existing methods. Specifically, we construct a biochemical KG for proteins and ligands, respectively. The encoded representations of proteins and ligands are integrated into the KG embedding objective to capture the rich semantic information within KG. Subsequently, the model employs cross attention mechanisms to perform fine-grained aggregation of the knowledge-enhanced representations, which are then fed into a decoder to predict binding affinity. We conduct extensive experiments to evaluate the effectiveness of KEPLA in addressing the performance and interpretability challenges.

In future work, we plan to incorporate 3D structural information into our PLA prediction models, leveraging recent advancements like AlphaFold [48], which has predicted over 2 billion protein structures. This integration has the potential to further improve model performance and interpretability.

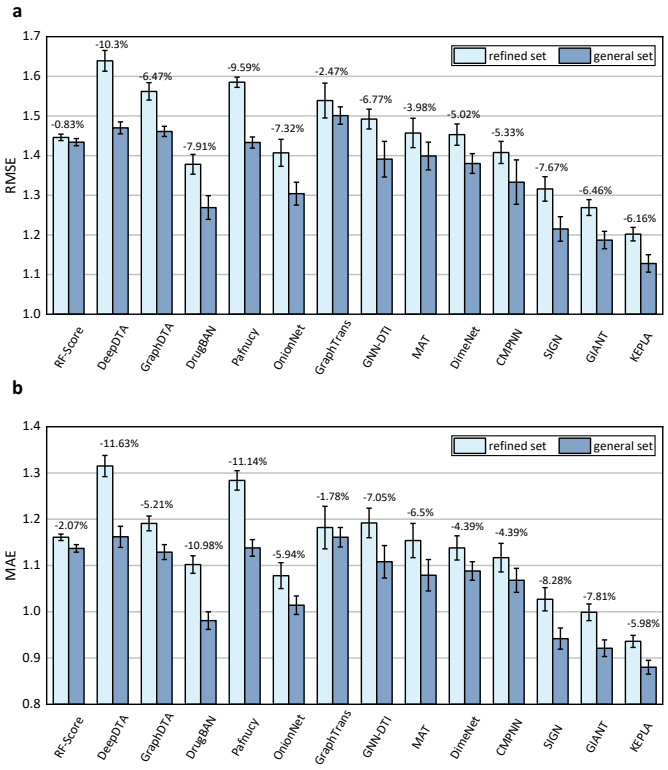


Fig. 6. **Performance improvements on PDBbind benchmark when trained on the *general set*. (a)** RMSE improvements on PDBbind core set. **(b)** MAE improvements on PDBbind core set.

Additionally, expanding our knowledge graph with more comprehensive domain-specific and cross-disciplinary knowledge could enhance prediction accuracy. We also aim to adapt the KEPLA framework to other molecular interaction prediction tasks, such as drug-drug and protein-protein interactions [31], [49], demonstrating its versatility and broader applicability.

References

[1] M. Volkov, J.-A. Turk, N. Drizard, N. Martin, B. Hoffmann, Y. Gaston-Mathé, and D. Rognan, "On the frustration to predict binding affinities from protein–ligand structures with deep neural networks," *Journal of Medicinal Chemistry*, vol. 65, no. 11, pp. 7946–7958, 2022.

[2] A. V. Sadybekov and V. Katritch, "Computational approaches streamlining drug discovery," *Nature*, vol. 616, no. 7958, pp. 673–685, 2023.

[3] Y. Luo, X. Zhao, J. Zhou, J. Yang, Y. Zhang, W. Kuang, J. Peng, L. Chen, and J. Zeng, "A network integration approach for drug-target interaction prediction and computational drug repositioning from heterogeneous information," *Nature Communications*, vol. 8, no. 1, p. 573, 2017.

[4] M. A. Rezaei, Y. Li, D. Wu, X. Li, and C. Li, "Deep learning in drug design: protein-ligand binding affinity prediction," *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, vol. 19, no. 1, pp. 407–417, 2020.

[5] Y. Li, D. Zhou, G. Zheng, X. Li, D. Wu, and Y. Yuan, "Dyscore: a boosting scoring method with dynamic properties for identifying true binders and nonbinders in structure-based drug discovery," *Journal of Chemical Information and Modeling*, vol. 62, no. 22, pp. 5550–5567, 2022.

[6] M. Li, Y. Cao, X. Liu, and H. Ji, "Knowledge-enhanced and structure-enhanced representation learning for protein-ligand binding affinity prediction," *Pattern Recognition*, vol. 166, p. 111701, 2025.

[7] H. Lai, L. Wang, R. Qian, J. Huang, P. Zhou, G. Ye, F. Wu, F. Wu, X. Zeng, and W. Liu, "Interformer: an interaction-aware model for protein-ligand docking and affinity prediction," *Nature Communications*, vol. 15, no. 1, p. 10223, 2024.

[8] D. Weininger, "Smiles, a chemical language and information system. 1. introduction to methodology and encoding rules," *Journal of Chemical Information and Computer Sciences*, vol. 28, no. 1, pp. 31–36, 1988.

[9] H. Öztürk, A. Özgür, and E. Ozkirimli, "Deepdta: deep drug–target binding affinity prediction," *Bioinformatics*, vol. 34, no. 17, pp. i821–i829, 2018.

[10] I. Lee, J. Keum, and H. Nam, "Deepconv-dti: Prediction of drug-target interactions via deep learning with convolution on protein sequences," *PLoS Computational Biology*, vol. 15, no. 6, pp. 1–21, 2019.

[11] M. Jiang, Z. Li, S. Zhang, S. Wang, X. Wang, Q. Yuan, and Z. Wei, "Drug–target affinity prediction using graph neural network and contact maps," *RSC advances*, vol. 10, no. 35, pp. 20 701–20 712, 2020.

[12] Z. Yang, W. Zhong, L. Zhao, and C. Y.-C. Chen, "Mgraphdta: deep multiscale graph neural network for explainable drug–target binding affinity prediction," *Chemical Science*, vol. 13, no. 3, pp. 816–833, 2022.

[13] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," *International Conference on Neural Information Processing Systems*, pp. 1–11, 2017.

[14] L. Chen, X. Tan, D. Wang, F. Zhong, X. Liu, T. Yang, X. Luo, K. Chen, H. Jiang, and M. Zheng, "Transformercpi: improving compound–protein interaction prediction by sequence-based deep learning with self-attention mechanism and label reversal experiments," *Bioinformatics*, vol. 36, no. 16, pp. 4406–4414, 2020.

[15] K. Huang, C. Xiao, L. M. Glass, and J. Sun, "Moltrans: molecular interaction transformer for drug–target interaction prediction," *Bioinformatics*, vol. 37, no. 6, pp. 830–836, 2021.

[16] M. Ashburner, C. A. Ball, J. A. Blake, D. Botstein, H. Butler, J. M. Cherry, A. P. Davis, K. Dolinski, S. S. Dwight, J. T. Eppig *et al.*, "Gene ontology: tool for the unification of biology," *Nature Genetics*, vol. 25, no. 1, pp. 25–29, 2000.

[17] Q. Ye, C.-Y. Hsieh, Z. Yang, Y. Kang, J. Chen, D. Cao, S. He, and T. Hou, "A unified drug–target interaction prediction framework based on knowledge graph and recommendation system," *Nature Communications*, vol. 12, no. 1, p. 6775, 2021.

[18] Q. Wang, Z. Mao, B. Wang, and L. Guo, "Knowledge graph embedding: A survey of approaches and applications," *IEEE Transactions on Knowledge and Data Engineering*, vol. 29, no. 12, pp. 2724–2743, 2017.

[19] Z. Lin, H. Akin, R. Rao, B. Hie, Z. Zhu, W. Lu, N. Smetanin, R. Verkuil, O. Kabeli, Y. Shmueli *et al.*, "Evolutionary-scale prediction of atomic-level protein structure with a language model," *Science*, vol. 379, no. 6637, pp. 1123–1130, 2023.

[20] A. Mastropietro, G. Pasculli, and J. Bajorath, "Learning characteristics of graph neural networks predicting protein–ligand affinities," *Nature Machine Intelligence*, vol. 5, no. 12, pp. 1427–1436, 2023.

[21] R. Hou, H. Chang, B. Ma, S. Shan, and X. Chen, "Cross attention network for few-shot classification," *International Conference on Neural Information Processing Systems*, pp. 4003–4014, 2019.

[22] P. J. Ballester and J. B. Mitchell, "A machine learning approach to predicting protein–ligand binding affinity with applications to molecular docking," *Bioinformatics*, vol. 26, no. 9, pp. 1169–1175, 2010.

[23] T. Nguyen, H. Le, T. P. Quinn, T. Nguyen, T. D. Le, and S. Venkatesh, "Graphdta: predicting drug–target binding affinity with graph neural networks," *Bioinformatics*, vol. 37, no. 8, pp. 1140–1147, 2021.

[24] P. Bai, F. Miljković, B. John, and H. Lu, "Interpretable bilinear attention network with domain adaptation improves drug–target prediction," *Nature Machine Intelligence*, vol. 5, no. 2, pp. 126–136, 2023.

[25] L. Zheng, J. Fan, and Y. Mu, "Onionnet: a multiple-layer intermolecular-contact-based convolutional neural network for protein–ligand binding affinity prediction," *ACS Omega*, vol. 4, no. 14, pp. 15 956–15 965, 2019.

[26] J. Lim, S. Ryu, K. Park, Y. J. Choe, J. Ham, and W. Y. Kim, "Predicting drug–target interaction using a novel graph neural network with 3d structure-embedded graph representation," *Journal of Chemical Information and Modeling*, vol. 59, no. 9, pp. 3981–3988, 2019.

[27] K. Yang, K. Swanson, W. Jin, C. Coley, P. Eiden, H. Gao, A. Guzman-Perez, T. Hopper, B. Kelley, M. Mathea *et al.*, "Analyzing learned molecular representations for property prediction," *Journal of Chemical Information and Modeling*, vol. 59, no. 8, pp. 3370–3388, 2019.

[28] M. M. Stepniewska-Dziubinska, P. Zielenkiewicz, and P. Siedlecki, "Development and evaluation of a deep learning model for protein–ligand binding affinity prediction," *Bioinformatics*, vol. 34, no. 21, pp. 3666–3674, 2018.

[29] S. Li, J. Zhou, T. Xu, L. Huang, F. Wang, H. Xiong, W. Huang, D. Dou, and H. Xiong, "Structure-aware interactive graph neural networks for the prediction of protein-ligand binding affinity," *ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pp. 975–985, 2021.

[30] ——, "Giant: Protein-ligand binding affinity prediction via geometry-aware interactive graph neural network," *IEEE Transactions on Knowledge and Data Engineering*, vol. 36, no. 5, pp. 1991–2008, 2024.

[31] X. Su, Z. You, D. Huang, L. Wang, L. Wong, B. Ji, and B. Zhao, "Biomedical knowledge graph embedding with capsule network for multi-label drug-drug interaction prediction," *IEEE Transactions on Knowledge and Data Engineering*, vol. 35, no. 6, pp. 5640–5651, 2022.

[32] T. Ma, X. Lin, B. Song, P. S. Yu, and X. Zeng, "Kg-mtl: knowledge graph enhanced multi-task learning for molecular interaction," *IEEE Transactions on Knowledge and Data Engineering*, vol. 35, no. 7, pp. 7068–7081, 2022.

[33] M. Li, J. Zhou, J. Hu, W. Fan, Y. Zhang, Y. Gu, and G. Karypis, "Dgl-lifesci: An open-source toolkit for deep learning on graphs in life science," *ACS Omega*, vol. 6, no. 41, pp. 27 233–27 238, 2021.

[34] A. Bordes, N. Usunier, A. Garcia-Durán, J. Weston, and O. Yakhnenko, "Translating embeddings for modeling multi-relational data," *International Conference on Neural Information Processing Systems*, pp. 2787–2795, 2013.

[35] Z. Sun, Z.-H. Deng, J.-Y. Nie, and J. Tang, "Rotate: Knowledge graph embedding by relational rotation in complex space," *International Conference on Learning Representations*.

[36] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *International Conference on Learning Representations*, 2015.

[37] Z. Liu, Y. Li, L. Han, J. Li, J. Liu, Z. Zhao, W. Nie, Y. Liu, and R. Wang, "Pdb-wide collection of binding data: current status of the pdbbind database," *Bioinformatics*, vol. 31, no. 3, pp. 405–412, 2015.

[38] R. D. Smith, J. B. Dunbar Jr, P. M.-U. Ung, E. X. Esposito, C.-Y. Yang, S. Wang, and H. A. Carlson, "Csar benchmark exercise of 2010: combined evaluation across all submitted scoring functions," *Journal of Chemical Information and Modeling*, vol. 51, no. 9, pp. 2115–2131, 2011.

[39] T. Danel, P. Spurek, J. Tabor, M. Śmieja, Ł. Struski, A. Słowik, and Ł. Maziarka, "Spatial graph convolutional networks," *International Conference on Neural Information Processing*, pp. 668–675, 2020.

[40] Z. Wu, P. Jain, M. Wright, A. Mirhoseini, J. E. Gonzalez, and I. Stoica, "Representing long-range context for graph neural networks with global attention," *International Conference on Neural Information Processing Systems*, vol. 34, pp. 13 266–13 279, 2021.

[41] M. Liu, Z. Wang, and S. Ji, "Non-local graph neural networks," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 44, no. 12, pp. 10 270–10 276, 2021.

[42] Ł. Maziarka, T. Danel, S. Mucha, K. Rataj, J. Tabor, and S. Jastrzębski, "Molecule attention transformer," *arXiv preprint arXiv:2002.08264*, 2020.

[43] J. Gasteiger, J. Groß, and S. Günnemann, "Directional message passing for molecular graphs," *International Conference on Learning Representations*, 2020.

[44] Y. Song, S. Zheng, Z. Niu, Z.-H. Fu, Y. Lu, and Y. Yang, "Communicative representation learning on attributed molecular graphs," *International Joint Conference on Artificial Intelligence*, pp. 2831–2838, 2020.

[45] M. Tsubaki, K. Tomii, and J. Sese, "Compound–protein interaction prediction with end-to-end learning of neural networks for graphs and sequences," *Bioinformatics*, vol. 35, no. 2, pp. 309–318, 2019.

[46] H. M. Berman, J. Westbrook, Z. Feng, G. Gilliland, T. N. Bhat, H. Weissig, I. N. Shindyalov, and P. E. Bourne, "The protein data bank," *Nucleic Acids Research*, vol. 28, no. 1, pp. 235–242, 2000.

[47] L. Maveyraud and L. Mourey, "Protein x-ray crystallography and drug discovery," *Molecules*, vol. 25, no. 5, p. 1030, 2020.

[48] J. Abramson, J. Adler, J. Dunger, R. Evans, T. Green, A. Pritzel, O. Ronneberger, L. Willmore, A. J. Ballard, J. Bambrick *et al.*, "Accurate structure prediction of biomolecular interactions with alphafold 3," *Nature*, vol. 630, no. 8016, pp. 493–500, 2024.

[49] J. Ji, A. Zhang, C. Liu, X. Quan, and Z. Liu, "Survey: Functional module detection from protein-protein interaction networks," *IEEE Transactions on Knowledge and Data Engineering*, vol. 26, no. 2, pp. 261–277, 2012.