DBGT-PLA: Dual-Branch Graph–Transformer Fusion for Interpretable Protein–Ligand Affinity Prediction

Ying Wang[1], Jing Hu[1,*], Junlin Xu[1], Bo Li[1,*]

***Abstract*— Protein-ligand binding affinity prediction is critical for drug discovery, yet existing methods struggle to jointly model local atomic interactions and global contextual dependencies. To address this, we propose the Interpretable Dual-Branch Graph–Transformer framework for Protein–Ligand Affinity prediction (DBGT-PLA), a novel dual-branch architecture that integrates graph neural network (GNN) with a stability-enhanced Transformer equipped with learnable positional embeddings and a NaN-filtering mechanism that handles potential Not-a-Number (NaN) values arising from numerical instability or data preprocessing. We design a Gated Residual Learning (GRL) Fusion module that performs dimension-wise adaptive integration between local graph topology and global Transformer context. This mechanism enables multi-level feature coordination through a residual path, achieving biophysically consistent alignment between atomic-level interactions and global conformational dependencies. Furthermore, we introduce an edge-level Shapley attribution framework tailored to protein–ligand interaction graphs, quantifying contributions of chemical bonds (e.g., hydrophobic contacts) and non-covalent interactions. Experiments show DBGT-PLA reduces RMSE by 18.3% (from 1.522 to 1.244 on the Holdout Set 2019), outperforming state-of-the-art models. Crucially, our explainability module reveals that the ligand edges dominate affinity predictions, accounting for nearly 70%. This work not only advances predictive accuracy but also offers unprecedented, quantitative insights into interaction determinants, which can guide rational drug optimization.**

***Index Terms*— Protein–ligand affinity, XAI method, Graph neural network, Graph transformer.**

I. Introduction

THE interactions between proteins and small-molecule ligands lie at the heart of vital biological processes, including enzymatic catalysis and signal transduction [1]. In the realms of drug design and molecular simulation, drug compounds typically serve as ligands that bind selectively to target proteins, exerting therapeutic effects by modulating intermolecular forces [2]. Accurate prediction of binding affinity substantially improves the efficiency and success rate of drug discovery, reduces development costs, and provides reliable structural and dynamic insights for molecular simulations—ultimately advancing the frontier of precision medicine.

[1]College of Computer Science and Technology, Wuhan University of Science and Technology, Wuhan 430070, China.

[*]Corresponding author: Jing Hu(hujing@wust.edu.cn) and Bo Li(libo@wust.edu.cn)

Precise measurement of protein–ligand binding affinity is essential for deciphering biomolecular interactions and translating them into practical applications in drug development. Experimental techniques such as isothermal titration calorimetry [3], surface plasmon resonance [4], and enzyme-linked immunosorbent assay [5] are widely regarded as the gold standard for affinity quantification. Nonetheless, their high time and labor costs, technical complexity, and limited scalability significantly restrict their utility in high-throughput scenarios, as they can only accommodate a narrow subset of protein–ligand pairs. To overcome these constraints, computational methods have emerged as indispensable tools, enabling the rapid screening of vast compound libraries without the need for extensive experimental resources.

Approaches for predicting protein–ligand binding affinity fall broadly into three categories: physics-based methods, molecular docking, and machine learning techniques. Physics-based strategies, such as molecular dynamics simulations [6] and free energy simulations [7], provide atomic-level precision but are computationally prohibitive for large-scale applications. Molecular docking [8], [9], by simulating a ligand's optimal conformation within the binding site of a protein, offers high-throughput potential and computational efficiency; however, its reliance on simplified energy functions limits predictive accuracy. In contrast, machine learning methods [10], [11]—exemplified by models like RFscore [12], [13] and Pred-binding [14] achieve substantial gains in both speed and accuracy by learning binding patterns from empirical data. Yet, their performance is often hindered by dependence on handcrafted features and difficulty in capturing the full spectrum of complex molecular interactions.

In the rapidly evolving intersection of artificial intelligence and bioinformatics, deep learning has emerged as a powerful tool, drawing widespread attention for its ability to autonomously learn feature representations directly from raw input data without relying on domain-specific prior knowledge [15]. This characteristic has propelled deep learning to the forefront of current research in protein–ligand binding affinity prediction, where a wide range of models have been proposed. These models can be broadly categorized into two main classes based on the type of input data they utilize: sequence-based and structure-based approaches.

Sequence-based methods have provided novel perspectives in affinity prediction. One of the seminal models in this category, DeepDTA [16], laid the groundwork for further

advancements. Building upon it, DeepDTAF [17] integrates multiple sources of sequence information such as protein binding pocket sequences and SMILES representations of ligands to uncover latent patterns associated with binding affinity. This integration enables more nuanced feature extraction and improved predictive performance. However, the reliance on linear sequence data in these models inevitably leads to the omission of critical structural insights, which may limit the models' capacity to fully capture the biophysical nature of protein–ligand interactions. As a result, the accuracy and reliability of predictions can be adversely affected [18].

To overcome this limitation, structure-based models have been developed, where protein–ligand complexes are encoded as three-dimensional (3D) grids or molecular graphs. For instance, Pafnucy [19] transforms complexes into 3D grids for input into deep neural networks. FAST [20], and IGN [21] adopt graph-based representations, capturing topological and chemical features via graph neural networks. More recently, Geo-PLA [22] introduces a local–global structure-aware framework that integrates geometric equivariant graph neural networks with graph Transformers, enabling joint modeling of local spatial geometry and long-range structural dependencies within protein–ligand complexes. This line of work highlights the importance of combining geometric equivariance with global contextual modeling for affinity prediction.

In addition to whole-complex structural modeling, pocket-centered approaches have recently gained traction for binding affinity prediction. These methods explicitly focus on the geometric and chemical environments surrounding the binding site, allowing models to extract fine-grained interaction patterns that may be diluted in whole-protein representations. For instance, MMPD-DTA [23] introduces a novel pocket-drug graph for atomic-level interaction modeling, fused with protein and drug representations in a multimodal framework, while DeepTGIN [24] employs a hybrid multimodal architecture, using dual Transformer encoders for protein and pocket sequences alongside a Graph Isomorphism Network (GIN) for ligand topology. Although these approaches have demonstrated notable performance gains, they typically depend on accurate pocket extraction or additional preprocessing pipelines, which may introduce bias or propagate docking errors.

Despite these strengths, structure-based models still face several essential challenges. First, grid-based representations may obscure fine-grained atomic geometry, and graph-based encodings often fail to preserve interfacial physicochemical contexts. Second, a growing body of evidence shows that traditional message-passing GNNs have intrinsic limitations in modeling long-range dependencies and heterogeneous biochemical interactions. For example, LLM-DDI [25] demonstrates that information propagation in GNNs is constrained by local neighborhoods, making it difficult to integrate multi-hop or semantically diverse signals across distant regions of a biomedical graph. Similarly, transformer-powered graph learning frameworks such as TREE [26] show that attention mechanisms naturally capture global and cross-type relational patterns that standard GNNs struggle to express. Recent studies have also emphasized the role of interaction-based inductive bias in addressing these challenges. For example, EHIGN [27] explicitly models heterogeneous protein–ligand interactions and decomposes binding affinity into atom-level contributions, improving generalization under cold-start settings. However, such approaches primarily focus on architectural inductive bias, while providing limited analysis of how interaction-driven representations contribute to interpretability across different affinity regimes. These findings suggest that purely GNN-based architectures may overlook distal residue–ligand interactions, allosteric effects, or long-range electrostatic influences that are essential for accurate affinity prediction. Finally, most existing approaches still lack a principled mechanism for interpretability.

In this paper, we propose DBGT-PLA, a dual-branch GNN–Transformer hybrid architecture specifically designed for learning from protein–ligand interaction graphs. Our framework captures both local topological and global contextual dependencies through a gated residual fusion mechanism and integrates explainable AI (XAI) [28] components for improved interpretability. Our contributions are summarized as follows:

- We propose a novel dual-branch architecture (DBGT-PLA) that combines a Graph Neural Network (GNN) for encoding local structural information and a Transformer module for capturing long-range contextual interactions.
- We design a StableTransformerLayer that enhances the standard Transformer by incorporating learnable positional embeddings, value clipping, and NaN filtering mechanisms. These improvements significantly boost numerical stability and convergence on biomolecular graph data.
- We develop a hierarchical GRL Fusion mechanism that performs dimension-wise gated residual integration between GNN-derived local topology and Transformer-derived global context. Unlike conventional single-step concatenation, this design achieves multi-level adaptive fusion, aligning molecular substructures and system-level conformations in a biophysically interpretable way.
- We incorporate interpretable learning via Shapley-based edge attribution, enabling edge-level explanations of affinity predictions.

II. RELATED WORK

A. Transformers

The advancement of Natural Language Processing (NLP) has been profoundly shaped by the emergence of the Transformer-based encoder–decoder architecture—a milestone introduced by Vaswani et al. [29]. Since its inception, the Transformer has rapidly become the standard paradigm for sequence modeling in NLP. The influence of the Transformer architecture extends far beyond natural language. In computer vision, Transformers have been employed for effective image analysis and interpretation [30]. In bioinformatics, they have enabled deeper modeling of protein amino acid sequences [31], offering new avenues for understanding protein structure and function. In cheminformatics, Transformers have proven effective in representing and processing molecular SMILES strings [32], accelerating progress in drug discovery and other chemical informatics applications.

B. Graph Neural Networks

Graph Neural Networks (GNNs) have rapidly emerged as a powerful and versatile framework for modeling and analyzing graph-structured data, demonstrating remarkable performance across a wide array of applications. GNNs operate via iterative message passing, where each node updates its representation by aggregating features from its neighbors and associated edge attributes. Aggregation functions—such as summation, averaging, or attention—enable nodes to progressively encode both local and global contextual information. This iterative refinement yields node embeddings that capture rich structural and semantic patterns. This inherent capacity to model intricate relational patterns has propelled GNNs to the forefront of diverse domains including social network analysis, recommender systems, and bioinformatics. Extensive research efforts have sought to enhance GNN architectures, with notable advances including the development of more expressive and adaptable aggregation functions, as proposed by Murphy et al. [33], Seo et al. [34], and Chatzianastasis et al. [35]. Additionally, to effectively incorporate diverse local structures and higher-order neighborhood information, innovative fusion techniques have been introduced, exemplified by the works of Morris et al. [36], and Nikolentzos et al. [37].

C. Explainable Artificial Intelligence (XAI) Methods

In machine learning, the goal is to construct a mapping function $f(X) = f(X_1, ..., X_M)$ from observed features $X = X_1, ..., X_M$ to a target variable Y for prediction or explanation. In complex nonlinear models such as deep neural networks or ensembles, the internal workings of f are typically opaque, making feature importance estimation critical—especially for tabular data. Widely used XAI techniques include feature importance metrics, LIME (Local Interpretable Model-agnostic Explanations), and SHAP (SHapley Additive explanations). For instance, Joo et al. [38] used SHAP to evaluate the influence of various substances on heat deflection temperature (HDT), aiding material selection. Lai et al. [39] applied SHAP to assess input importance for the performance of concrete-filled steel tubes under lateral impact, providing interpretability and parametric insights. Similarly, Kashifi [40] employed SHAP to interpret a Gated Recurrent Convolutional Network (GRCN) in predicting spatiotemporal collision risks, identifying key risk factors that support effective safety interventions.

III. METHODS

In this section, we present our proposed framework, DBGT-PLA. An illustration of the proposed architecture can be found in Fig. 1. To provide a clear roadmap of our methodology, the framework consists of four integral modules: first, the Input Representation, where the 3D protein-ligand complex is transformed into an interaction graph to explicitly encode atomic spatial relationships; second, the Dual-Branch Feature Encoding, where the graph is processed in parallel by a GC-GNN module capturing local topological structures and a Stable Transformer module modeling long-range global dependencies; third, the GRL Fusion, which adaptively integrates these local and global features via a gated residual path; and finally, the Prediction & Interpretation module, which aggregates the fused features for affinity prediction and applies a Shapley-based method to elucidate specific interaction edges.

A. Protein–Ligand Interaction Graphs

For these complexes, interaction graphs provided by Volkov et al. [41] were publicly accessible, and representative samples were reconstructed based on distances and interaction types identified using IChem and NetworkX. The construction of these graphs strictly followed the original protocol for defining ligand and protein pseudo-atoms. Specifically, an edge was drawn between a ligand pseudo-atom and a protein pseudo-atom if their spatial distance was less than 6 Å. This threshold was deliberately chosen: since protein residues are simplified as pseudo-atoms in this study, a 6 Å cutoff substantially increases the number of interaction edges compared to a 4 Å threshold.Empirically, interaction graphs built with this 6 Å cutoff yielded lower root mean square error (RMSE) in prediction tasks, demonstrating improved predictive accuracy.Similarly, when the distance between two ligand pseudo-atoms or between two protein pseudo-atoms was under 4 Å—accounting for covalent and non-covalent intramolecular interactions—edges were also established. Edge annotations represent interaction distances, which were standardized and scaled using the interquartile range of the global distance distribution rather than raw measurements. This normalization was performed with the RobustScaler function from Scikit-learn.Node features were encoded using a one-hot scheme, where each entry signifies a specific type of non-covalent interaction. A value of 1 indicates that the pseudo-atom participates in the corresponding interaction. The interaction types include: CA for hydrophobic interactions; NZ for ionic interactions with positively charged protein atoms; N for hydrogen bonds where the atom acts as a donor; OG for atoms serving as both hydrogen bond donors and acceptors; O for hydrogen bond acceptors only; CZ for aromatic interactions; OD1 for ionic interactions with negatively charged protein atoms; and ZN for metal coordination interactions.

B. GNN Architectecture

The GC-GNN component is implemented as a modular graph neural network using the PyTorch Geometric framework. The model architecture employs the GraphConv operator, which is derived from the innovative graph convolution method proposed by Morris et al. [42]. The network consists of a configurable number of GraphConv layers (default: 7 layers). Each GraphConv layer contains 256 hidden units, consistent with the hidden dimension specified in our model architecture parameters. Following each convolutional operation, a Rectified Linear Unit (ReLU) activation function is applied to enhance the model's nonlinear representational capacity. To mitigate overfitting during training, a dropout layer with a rate of 0.5 is incorporated after each GraphConv layer, randomly deactivating a proportion of neurons to improve generalization.The forward propagation processes node features through the sequential GraphConv layers, where each

layer transforms and refines the node representations. The model outputs node-level embeddings that are subsequently utilized by downstream components for graph-level prediction tasks. This modular design allows for flexible configuration of network depth while maintaining consistent hidden dimensions throughout the architecture.

The mathematical formulation of the GraphConv operator is defined as follows:

$$X_i^{'} = W_1 X_i + W_2 \max_{j \in \mathcal{N}(i)} \left(e_{j,i} X_j \right) \tag{1}$$

where W_1 and W_2 are the neural network weights, $e_{j,i}$ represents the edge weight from node j to node i, X_i and X_j are the feature vectors for nodes i and j, respectively, and $\mathcal{N}(i)$ denotes the set of neighboring nodes of node i.

C. Transformer Module

To effectively capture global contextual dependencies within protein-ligand interaction graphs, we design a robust Transformer-based architecture tailored for biomolecular data. While standard Transformers excel in sequence modeling, their direct application to heterogeneous molecular graphs often encounters numerical instability due to the dynamic scaling of feature representations during the training process. Specifically, in complex binding pockets, the dot-product attention scores can exhibit extreme variance, pushing the Softmax function into saturation regions or leading to gradient explosion.

To address this, we propose the StableTransformerLayer, a specialized block that incorporates specific architectural and numerical enhancements. Each layer comprises a multi-head self-attention mechanism and a feed-forward network employing GELU activation for smoother gradient propagation. We introduce a learnable positional bias (parameterized as a broadcastable vector) that is added to the node features at the input of each layer. This mechanism allows the model to adaptively refine the spatial identity of nodes locally within each depth of the network, which is critical for distinguishing biophysical features in graph-structured data.

Crucially, to ensure convergence, we implement a Runtime Stabilization Mechanism aligned with the computation flow:

- Input/Output Anomaly Filtering: Given the sparse nature of interaction graphs, we apply `torch.nan_to_num` at both the entry and exit of the layer. This proactively neutralizes any NaN or infinite values that may arise from division-by-zero errors in upstream sparse aggregations or attention computations.
- Post-Normalization Clipping: We observe that standard Layer Normalization can occasionally amplify numerical instability when feature variance is minimal. Therefore, we strictly enforce value clipping via `torch.clamp` immediately after each normalization step (both after the attention block and the feed-forward block). This confines the feature manifold within a stable range, preventing the propagation of extreme values to subsequent layers.

Collectively, these modifications transform the standard Transformer block into a stable encoder capable of handling the heterogeneous distribution of protein-ligand affinity data.

D. GRL Fusion Module

To integrate local and global features effectively, we introduce the GRL Fusion module. While GNN-derived and Transformer-derived features offer complementary perspectives, naive fusion (e.g., simple addition or concatenation) may result in suboptimal representation due to feature scale imbalance or contextual mismatch. GRL Fusion module addresses this by applying a dimension-wise gating mechanism that adaptively modulates the Transformer output before merging it with the GNN features via a residual connection. The architecture and functionality are described as follows. The GRL Fusion module consists of a Multi-Layer Perceptron (MLP) comprising two linear transformations and a non-linear activation function (ReLU), followed by a Sigmoid activation function to generate gating weights.

Let g denote the GNN-derived feature and t the Transformer-derived feature,where $g, t \in \mathbb{R}^d$ and d is the feature dimension. These two feature vectors are first concatenated to form a joint representation $c = [g; t] \in \mathbb{R}^{2d}$. The MLP then computes a gating weight vector g_w as follows:

$$g_w = Sigmoid(MLP(c)) \tag{2}$$

The MLP begins by projecting the 2d-dimensional input into a d-dimensional latent space through a linear layer, followed by a ReLU activation to introduce non-linearity. A second linear layer maintains the output dimensionality at d, after which a Sigmoid activation is applied to constrain the gating weights to the range [0,1]. Finally, the fused feature vector f is computed using the following formula:

$$f = g + g_w \cdot t \tag{3}$$

This residual formulation offers two practical advantages over direct concatenation or naive addition. First, the learnable gate g_w adaptively rescales the Transformer-derived global features before fusion, effectively preventing feature-scale imbalance between the two branches. Second, by preserving g as the residual backbone and injecting only gated global information, the model maintains stable local structural representations while selectively enhancing them with complementary long-range context.

From a biophysical perspective, this design mimics the nature of protein–ligand interactions, where local atomic environments determine binding specificity, while global conformational context modulates the overall binding stability. By realizing adaptive local–global balance through GRL, the model achieves biophysically consistent multi-level fusion, effectively bridging atomic-scale precision and system-level coherence.

E. Model Interpretability Module

In XAI analysis, the mechanism for representing information in interaction graphs is a critical consideration. Typically, such graphs are structured as topologies composed of ligand pseudo-atoms (node type 1) and protein pseudo-atoms (node type 2) serving as fundamental units. Ligand pseudo-atoms represent ligand atoms or intermediate positions between atoms forming intramolecular non-covalent

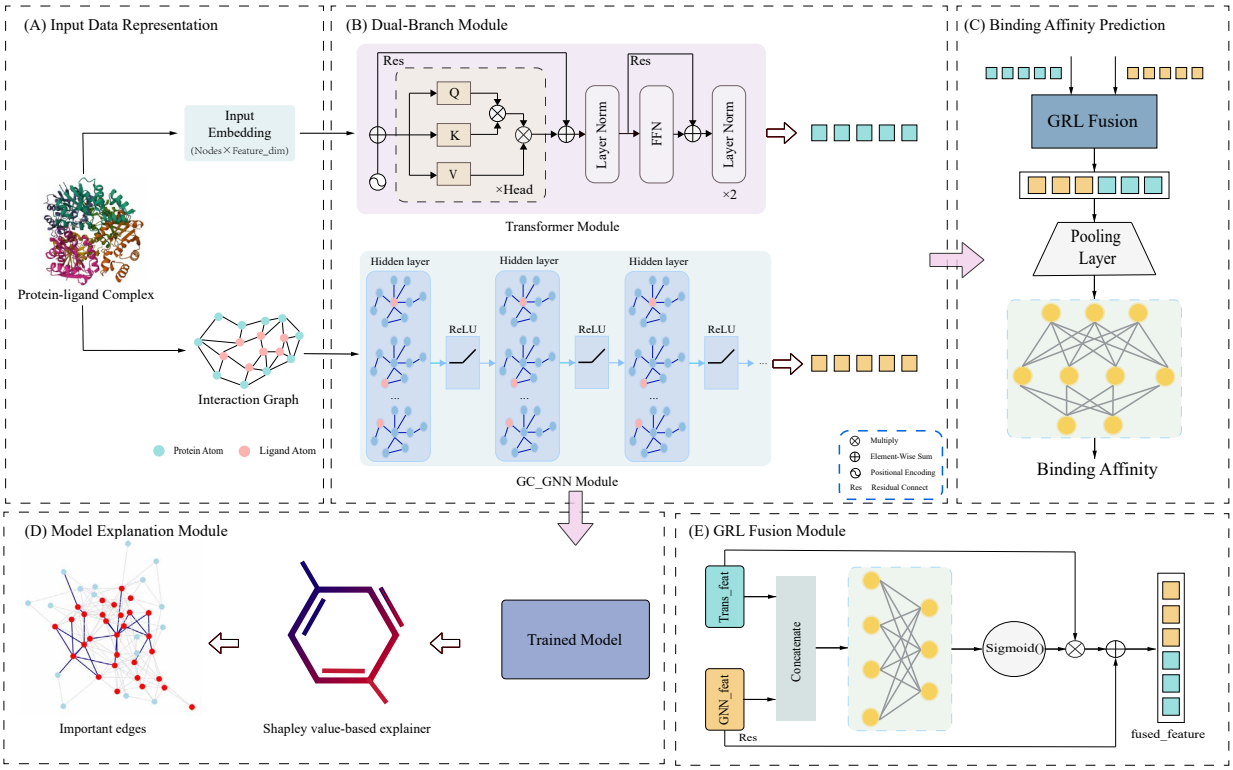

Fig. 1. Architectural framework of DBGT-PLA. The model takes a protein–ligand complex as input and first transforms its structural information into an interaction graph represented by nodes. Node features are then encoded in parallel via a dual-branch architecture: one branch leverages a GNN to capture local topological information of the graph, while the other employs a Transformer to model global contextual dependencies. The two sets of features are subsequently integrated using the Gated Residual Learning (GRL) module, which performs dimension-wise weighted fusion via a gating mechanism. This enables the model to adaptively balance the contributions of local and global representations. The fused node features are then aggregated from multiple perspectives using three graph-level pooling strategies—Add, Max, and Mean—providing a comprehensive summary of the graph. The aggregated features are passed through a fully connected layer to predict the binding affinity. Finally, the model identifies the importance of edges contributing to the prediction and highlights subgraph structures that are most influential in the decision-making process.

interactions and protein pseudo-atoms represent amino acid residues.Interaction edges account for different types of non-covalent intermolecular interactions (hydrogen bonds or hydrophobic and/or van der Waals interactions).

To satisfy the interpretability requirements of GNN predictions, this study employs ShapG [43], a model-agnostic global explanation tool that quantifies the importance of edges learned by GNNs and identifies those most critical to individual predictions. Initially, ShapG constructs an undirected graph from the dataset, with nodes representing features and edges established based on correlation coefficients between features. Subsequently, it estimates approximate Shapley values by sampling the data in the context of this graph structure. Originally devised to quantify each player's contribution to a team's success, Shapley values in machine learning map players to features—such as graph edges—and games to individual prediction tasks. ShapG thus ranks molecular graph edges by their contribution to specific predictions.

F. Loss Function

We propose a Composite Loss Function for protein–ligand binding affinity prediction that integrates the strengths of multiple loss terms to enhance the model's predictive accuracy. The Composite Loss Function is formulated as follows,

$$\mathcal{L} = \alpha \mathcal{L}_{Huber} + \beta \mathcal{L}_{PCC} \tag{4}$$

Here, α and β are weighting coefficients that balance the contributions of the Huber loss and Pearson correlation loss.

To evaluate the robustness of the Composite Loss Function, we conducted a sensitivity analysis on the PDBBind 2019 validation by varying the trade-off parameters α and β. The total loss weight was constrained to $\alpha + \beta = 1.0$, with α gradually increased from 0 to 1.0 and β adjusted accordingly. As shown in Fig. 2, the model achieved its best validation performance when $\alpha = 0.3$ and $\beta = 0.7$; hence,this configuration was adopted as the default setting in all subsequent experiments.

The Huber loss, which combines the benefits of Mean Squared Error (MSE) and Mean Absolute Error (MAE), applies a quadratic penalty to small residuals and a linear penalty to large residuals, thereby providing robustness against outliers. It is defined as,

$$\mathcal{L}_{Huber} = \begin{cases} 0.5\,(y-\hat{y})^2, & |y-\hat{y}| \le \delta \\ \delta\,(|y-\hat{y}| - 0.5\delta), & \text{otherwise} \end{cases} \tag{5}$$

where δ is the threshold separating quadratic and linear regimes, set to $\delta = 1$.

The Pearson correlation loss quantifies the linear association between predicted and true values, defined as one minus the

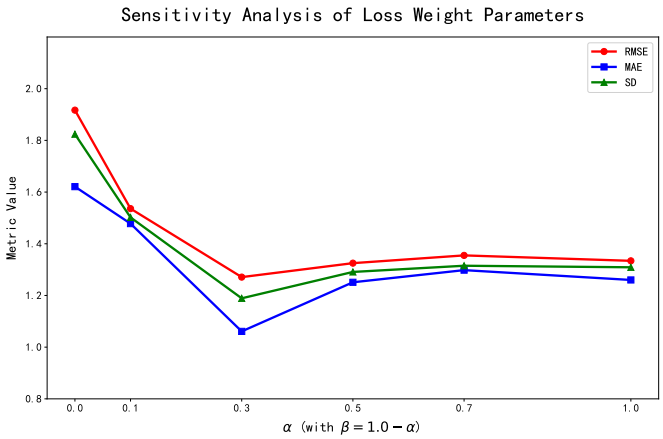

Fig. 2. Sensitivity analysis of the loss function parameters α and β (α + β = 1.0). Plots show RMSE, MAE and SD on the PDBBind 2019 validation set as α varies from 0.0 to 1.0.

Pearson correlation coefficient:

$$\mathcal{L}_{PCC} = 1 - \frac{\left(\sum(y_{\text{pred}} - \bar{y}_{\text{pred}})(y_{\text{true}} - \bar{y}_{\text{true}})\right)^2}{\sum(y_{\text{pred}} - \bar{y}_{\text{pred}})^2 \cdot \sum(y_{\text{true}} - \bar{y}_{\text{true}})^2} \tag{6}$$

where $\bar{y}_{\text{pred}}$ and $\bar{y}_{\text{true}}$ denote the means of predicted and true values, respectively.

G. Training Strategy and Optimization

To ensure robust model performance and generalization in protein-ligand affinity prediction, we employed several training strategies to enhance model stability and generalization. First, we used a linear learning rate warmup for the first 10 epochs, gradually increasing the rate from 10% to 100% of the initial value (1×10^{-3}), followed by a `ReduceLROnPlateau` scheduler that halved the learning rate if the validation RMSE did not improve for 5 epochs. To prevent overfitting, we incorporated L2 weight decay with a coefficient of 0.01 and applied gradient clipping with a maximum L2 norm of 1.0. Early stopping was triggered if no improvement was observed for 15 epochs. Additionally, we reset the learning rate to half its initial value (5×10^{-4}) if it dropped below 1×10^{-5} to avoid training stagnation. The best model was saved based on the lowest validation RMSE.

These strategies collectively addressed the challenges of training deep learning models on protein-ligand graph data, such as gradient instability and overfitting, leading to consistent and reliable results.

IV. RESULTS

A. Experimental Setups

1) Datasets: In this study, the training and validation datasets were derived from the PDBbind v2019 dataset. The training set comprises 9,962 protein–ligand complexes with corresponding binding affinity data, while the validation set includes 903 complexes with affinity information. For testing, the 2013 core, 2016 core and 2019 holdout sets from PDBbind were selected. Following data retrieval and curation, these sets contain 195, 257 and 3393 protein–ligand complexes, respectively, all obtained from the official PDBbind website `http://www.pdbbind.org.cn`. To ensure data quality and reliability, stringent filtering was applied across all datasets. Complexes exhibiting anomalous affinity values—specifically, potency below 5 pK_i or above 11 pK_i (where pK_i denotes the negative logarithm of potency)—were excluded. Additionally, complexes failing interaction graph construction due to structural ambiguities were removed to maintain the integrity of subsequent analyses. After this rigorous preprocessing, the finalized datasets comprised 7,301 complexes in the training set, 658 complexes in the validation set, 111 complexes in the 2013 core test set, 186 complexes in the 2016 core test set, and 2542 complexes in the 2019 holdout test set.

TABLE I
KEY HYPERPARAMETER SETTINGS FOR DBGT-PLA

Category	Parameter	Value
Training	Epochs	100
	Batch size	32
	Optimizer	Adam
	Learning rate	1×10^{-3}
	Weight decay	5×10^{-4}
Model Architecture	Hidden channels	256
	GNN layers	7
	Transformer layers	2
	Attention heads	4
Regularization	Dropout (Transformer)	0.1
	Dropout (GNN)	0.5
	Gradient clipping	1.0 (L2 norm)

To stabilize the training process and facilitate optimization, the affinity values were standardized using the mean and variance estimated from the training set. The same transformation was applied to the validation and test sets, while their original values were recovered by inverse transformation before evaluation.

2) Hyperparameter settings: The hyperparameters in DBGT-PLA are listed in Table I. We employed a consistent set of hyperparameters across all datasets to ensure fair comparison and reproducibility. The training protocol used the Adam optimizer with a learning rate of 1×10^{-3} and weight decay of 5×10^{-4}, with a batch size of 32 and 100 training epochs. The model architecture featured a hidden channel dimension of 256, with 2 transformer layers (4 attention heads each) and 7 GNN layers. To prevent overfitting, we applied dropout rates of 0.1 and 0.5 for the transformer and GNN components, respectively, along with gradient clipping at an L2 norm of 1.0.

3) Evaluation Metrics: The model's predictive performance on protein–ligand binding affinity was evaluated using multiple metrics, including Root Mean Square Error (RMSE), Mean Absolute Error (MAE), and Standard Deviation (SD), each offering distinct insights into the model's accuracy.

Root Mean Square Error (RMSE) is a standard measure of the difference between predicted and true values. It calculates the average of the squared errors between the predicted and true values and then takes the square root. The smaller the RMSE, the more accurate the prediction of the model is. The formula is as follows.

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(Y_i - \hat{Y}_i)^2} \tag{7}$$

TABLE II
COMPARISON RESULTS OF DBGT-PLA AND BASELINE MODELS ON THREE TEST SETS.

Model	Core set 2013			Core set 2016			Holdout set 2019		
	RMSE ↓	MAE ↓	SD ↓	RMSE ↓	MAE ↓	SD ↓	RMSE ↓	MAE ↓	SD ↓
DeepDTA	1.436(0.021)	1.369(0.016)	1.415(0.035)	1.414(0.031)	1.167(0.015)	1.410(0.019)	1.522(0.010)	1.391(0.019)	1.504(0.015)
Pafnucy	1.478(0.019)	1.351(0.011)	1.452(0.015)	1.480(0.014)	1.308(0.027)	1.464(0.022)	1.471(0.015)	1.370(0.018)	1.455(0.021)
IGN	1.399(0.032)	1.203(0.021)	1.396(0.030)	1.396(0.035)	1.137(0.021)	1.376(0.014)	1.403(0.027)	1.265(0.031)	1.396(0.027)
GIGN	1.343(0.010)	1.187(0.024)	1.331(0.018)	1.294(0.010)	1.121(0.023)	1.285(0.012)	1.368(0.026)	1.152(0.010)	1.302(0.024)
CAPLA	1.371(0.021)	1.195(0.028)	1.355(0.011)	1.312(0.021)	1.129(0.019)	1.291(0.026)	1.367(0.016)	1.200(0.029)	1.294(0.012)
EHIGN	1.264(0.034)	1.137(0.032)	1.260(0.019)	1.259(0.042)	1.114(0.024)	1.242(0.025)	1.386(0.023)	1.202(0.019)	1.307(0.009)
DEAttentionDTA	1.259(0.040)	1.113(0.031)	1.242(0.016)	1.242(0.017)	1.110(0.012)	1.235(0.015)	1.296(0.009)	1.173(0.024)	1.283(0.011)
GNNSeq	1.253(0.025)	1.098(0.013)	1.249(0.030)	1.249(0.021)	1.103(0.017)	1.231(0.025)	1.287(0.033)	1.192(0.039)	1.277(0.020)
DBGT-PLA	**1.237(0.012)**	**1.068(0.009)**	**1.231(0.010)**	**1.231(0.006)**	**1.096(0.009)**	**1.225(0.007)**	**1.244(0.016)**	**1.024(0.016)**	**1.218(0.007)**

Mean Absolute Error (MAE) is the average of the absolute differences between the predicted and true values. The smaller the MAE, the more accurate the prediction of the model is. The formula is as follows.

$$\mathrm{MAE} = \frac{1}{n}\sum_{i=1}^{n}|Y_i - \hat{Y}_i| \tag{8}$$

Standard Deviation (SD) is a measure of the dispersion of a data distribution or variable. Standard deviation is often used to evaluate the stability of model predictions. If the standard deviation is small, it means that the prediction results of the model are relatively stable; otherwise, it means that the prediction results fluctuate greatly. The formula is as follows.

$$\mathrm{SD} = \sqrt{\frac{1}{n-1}\sum_{i=1}^{n}[y_i - (a\cdot p_i + b)]^2} \tag{9}$$

where n is the number of protein–ligand pairs, y_i is the experimental affinity, p_i is the estimated value of the i-th pair, and a and b are the slope and intercept of the function line between the ground truth values and predicted values.

4) Implementation Details and Reproducibility: To ensure the reliability and reproducibility of our experimental results, all models were implemented using the PyTorch framework. We conducted five independent training and evaluation runs for each model. Each run was initialized with a predefined set of random seeds (42, 123, 456, 789, 1000) to control stochastic factors such as model parameter initialization and data loading order. All performance metrics reported in this paper are the arithmetic mean of the results from these five runs, with the standard deviation indicated in parentheses (e.g., RMSE: 1.231 ± 0.006) to quantitatively assess the model's stability.

B. Model Comparison

To evaluate the predictive performance of DBGT-PLA in the protein–ligand binding affinity prediction, we compared our model with multiple existing methods on the constructed datasets. The compared methods include DeepDTA [16], Pafnucy [19], IGN [21], GIGN [44], CAPLA [45], DEAttentionDTA [46], EHIGN [27] and GNNSeq [47]. We retrained and reevaluated the prediction performance of these methods, respectively. The root mean square error (RMSE), mean absolute error (MAE), and standard deviation (SD) in regression are used to measure the predictive performance. As shown in Table II, DBGT-PLA consistently achieves the best performance across all three metrics on all test datasets.

On the Core set 2013, DBGT-PLA establishes its superior performance at the outset, achieving the lowest RMSE (1.237 ± 0.012), MAE (1.068 ± 0.009), and SD (1.231 ± 0.010) among all evaluated models. It marginally outperforms the strong baseline GNNSeq (RMSE: 1.253, MAE: 1.098), demonstrating a consistent advantage even on earlier benchmark data.

On the Core set 2016, DBGT-PLA maintains this leading performance, obtaining the lowest RMSE (1.231 ± 0.006) and MAE (1.096 ± 0.009), outperforming other strong baselines such as DEAttentionDTA (RMSE: 1.242, MAE: 1.110) and EHIGN (RMSE: 1.259, MAE: 1.114).

Most notably, on the more challenging Holdout set 2019, DBGT-PLA continues to surpass all competing models, achieving an RMSE of 1.244 (±0.016), MAE of 1.024 (±0.016), and SD of 1.218 (±0.007). Compared with DEAttentionDTA (RMSE: 1.296, MAE: 1.173) and EHIGN (RMSE: 1.386, MAE: 1.202), DBGT-PLA reduces RMSE by 4.0–10.2% and MAE by 12–15%. Furthermore, the low standard deviations (in parentheses) reported for DBGT-PLA across all metrics and datasets underscore the robustness and reliability of our method compared to other approaches.

The superior performance of DBGT-PLA across all benchmark sets demonstrates its strong capability in predicting protein-ligand binding affinity. To further reinforce that this performance stems from learning genuine structure-affinity relationships rather than dataset-specific artifacts, we conducted a critical sanity check: a control experiment with randomly assigned affinity values. This experiment followed the identical training pipeline and hyperparameters, with the only alteration being the replacement of true affinity labels with values randomly sampled from a uniform distribution matching the original value range.

The results of this experiment are unequivocal: the model's performance degrades drastically when the meaningful structure-label correlation is broken. As quantitatively detailed in Table III, the spikes in RMSE, MAE, and SD across the 2013 core, 2016 core and 2019 holdout sets confirm that the model cannot generalize or perform well without learning authentic biomolecular features.

This stark contrast provides compelling evidence that DBGT-PLA's predictive power is robust and valid. It conclusively rules out the possibility that the model's high accuracy

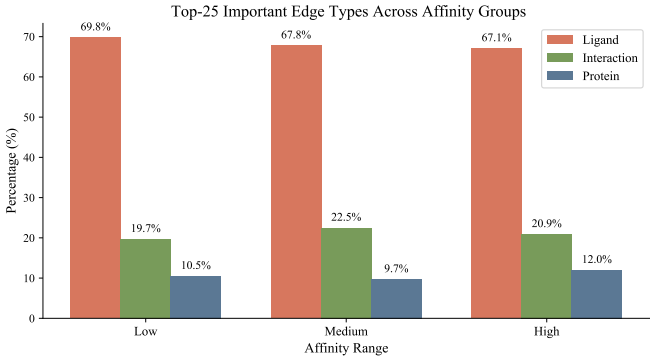

Fig. 3. Relative proportions of edges determining predictions for different affinity subranges. Colour-coded bars compare mean proportions of protein, ligand and interaction edges among the top 25 edges determining test set predictions prioritized by ShapG.

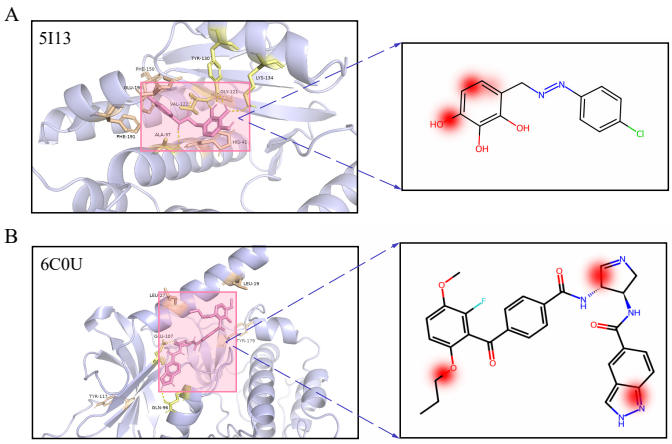

Fig. 4. Structural validation of ShapG attributions on two representative complexes. (A) PDB ID 5I13; (B) PDB ID 6C0U. Left panels: PyMOL rendering of the protein binding pocket and the docked/co-crystal ligand pose; pocket residues near the ligand are labeled. Right panels: ShapG edge-level attributions mapped onto the 2D ligand structure; warmer colors indicate higher positive contributions to the predicted binding affinity.

was achieved by overfitting to label noise or exploiting trivial patterns in the data. Instead, it underscores that our architecture successfully captures the underlying biophysical principles governing protein-ligand interactions, thereby solidifying the claims of its superiority established in the model comparisons above.

Taken together, these results demonstrate that DBGT-PLA exhibits strong robustness and excellent predictive capability across multiple test sets, confirming its effectiveness in the task of protein–ligand binding affinity prediction.

TABLE III
COMPARISON OF METRICS UNDER ORIGINAL VS. RANDOM AFFINITIES.

Dataset	Setting	RMSE↓	MAE↓	SD↓
core set 2013	Original Affinities	1.237	1.068	1.231
core set 2013	Random Affinities	1.718	1.461	1.716
core set 2016	Original Affinities	1.231	1.096	1.225
core set 2016	Random Affinities	1.745	1.502	1.743
holdout set 2019	Original Affinities	1.244	1.024	1.218
holdout set 2019	Random Affinities	1.743	1.509	1.740

C. Interpretability Results

For each prediction, the k most important edges were identified using ShapG (Methods), that is, the top k edges with the highest absolute SHAP values. Edges are classified into three different categories including (1) intramolecular edges formed between ligand pseudo-atoms (termed ligand edges), (2) intramolecular edges formed between protein pseudo-atoms (protein edges) and (3) intermolecular edges formed between ligand and protein pseudo-atoms (interaction edges). During the model interpretability analysis, we conducted a series of evaluations using the core set 2016 as the test dataset. Specifically, we examined the model's predictive performance across different binding affinity subranges, including the low-affinity range ($pK_i < 6$), the medium-affinity range($pK_i \in [6.5, 7.5]$), and the high-affinity range ($pK_i > 8$). To ensure clear separation among these affinity intervals and to avoid potential boundary effects arising from the continuous nature of affinity values, we adopted the following strategy: All test instances falling into the intermediate regions—i.e., those within 0.5 pK_i units between the defined subranges—were excluded from the analysis. After this filtering process, the low-affinity subrange contained 41 test instances, the medium-affinity subrange included 49 instances, and the high-affinity subrange comprised 62 instances.

Fig. 3 compares the proportions of protein, ligand, and interaction edges among the top 25 ranked edges, highlighting their respective contributions to the prediction of binding affinity. As shown in Fig. 3, ligand edges consistently dominate the top 25 most important edges across all three binding affinity subranges, accounting for approximately 70%, 68%, and 67% in the low, medium, and high affinity groups, respectively. This consistent pattern indicates that the model primarily relies on intraligand structural information during prediction. This aligns with empirical observations in medicinal chemistry, where ligand intramolecular flexibility often governs entropic contributions to binding free energy. In contrast, the contribution of interaction edges ranges from 19.7% to 22.5%, showing a slight increase in relevance within the medium and high affinity groups. This suggests that intermolecular interactions play a more prominent role in predictions when binding is stronger—likely due to the involvement of more stable or specific protein–ligand contacts. The contribution of protein edges is substantially smaller, representing only about 11% of the top 25 edges across all affinity subranges. This might be due to the much lower propensity of protein edges than ligand edges in the graphs.

To further investigate the interpretability of our model, we applied the ShapG method to visualize the edge-level Shapley values for representative complexes (PDB IDs: 5I13 and 6C0U). This analysis highlights the substructures of ligands that contribute most to the binding affinity predictions. For structural validation of ShapG attributions, we generated binding poses using molecular docking and visualized pocket contacts in PyMOL. For each protein–ligand complex, ligand conformers were docked into the experimental binding site

using AutoDock4 with default parameters; finally the docked pose closest to the co-crystal conformation was selected and visualized.

In the complex of the Endonuclease inhibitor 2 bound to the influenza strain H1N1 polymerase acidic subunit N-terminal region at pH 7.0 (PDB ID: 5I13), as shown in Fig. 4A, nine residues within the binding pocket—including Ala37, Val122, Tyr130, Lys134, His41, Gly121, Phe150, Phe191, and Glu195—were successfully identified. Specifically, Ala37, Val122, Tyr130, and Lys134 form hydrogen bonds with the ligand, while the other residues engage in hydrophobic interactions with the remaining atoms of the compound. Notably, the ShapG interpretability analysis strongly emphasized the ligand's hydroxyl groups (–OH) and the adjacent aromatic ring system. These substructures are chemically significant: the hydroxyl groups act as critical hydrogen bond donors or acceptors, directly facilitating the observed polar interactions with Ala37, Val122, Tyr130, and Lys134. Concurrently, the conjugated aromatic system enhances binding stability through hydrophobic contacts and potential π-π stacking within the pocket.

As depicted in Fig. 4B, for the complex of the cAMP-dependent protein kinase Calpha subunit with ligand N46 (PDB ID: 6C0U), the model accurately localizes the binding interface by identifying six pivotal residues: Leu19, Leu27, Gln96, Glu107, Tyr117, and Tyr179. Unlike the broad interaction patterns observed in other complexes, the model here correctly differentiates the specific hydrogen-bonding anchors (Gln96 and Glu107) from the supporting hydrophobic network mediated by the remaining residues. From an interpretability perspective, the ShapG visualization reveals that the model's attention is not uniformly distributed but rather concentrated on the indazole scaffold and the central amide-pyrrolidine linker. This focus is biologically profound: the indazole moiety functions as a critical 'hinge-binder,' mimicking the adenine ring of ATP to lock the ligand into the active site via hydrogen bonds with the hinge residue Glu107. By prioritizing these rigid, orientation-determining substructures over the flexible peripheral alkyl chains, the model demonstrates an ability to discern the essential pharmacophoric features required for potent kinase inhibition.

Collectively, these case studies demonstrate that our model not only achieves accurate affinity predictions but also autonomously identifies chemically meaningful pharmacophores consistent with biological mechanisms. By pinpointing specific binding determinants—ranging from the polar transition-state mimics in 5I13 to the rigid hinge-binding scaffolds in 6C0U—the ShapG-based visualization effectively bridges the gap between deep learning representations and established principles of medicinal chemistry. The model's distinctive ability to prioritize essential interaction motifs over peripheral flexible chains suggests an implicit understanding of biophysical constraints. Consequently, this interpretable framework serves as a reliable tool for structure-based drug design, guiding medicinal chemists to focus optimization efforts on functionally critical substructures rather than non-essential peripheral components.

TABLE IV
ABLATION STUDY RESULTS ON THE CORE SET 2013.

Models	G	T	F	C	RMSE↓	MAE↓	SD↓
DBGT-PLA-A	✓	×	×	×	2.691	2.180	1.995
DBGT-PLA-B	✓	×	×	✓	2.427	1.998	1.901
DBGT-PLA-C	✓	✓	×	×	1.790	1.437	1.652
DBGT-PLA-D	✓	✓	✓	×	1.480	1.285	1.416
DBGT-PLA-E(no-res.)	✓	✓	✓	×	1.492	1.381	1.487
DBGT-PLA-F(std.-T)	✓	✓	✓	×	1.503	1.389	1.471
DBGT-PLA	✓	✓	✓	✓	**1.237**	**1.068**	**1.231**

G: GC-GNN; T: Transformer (stable);
F: GRL Fusion; C: Composite Loss.
"No-res.": GRL without residual. "Std.-T": standard Transformer.

TABLE V
ABLATION STUDY RESULTS ON THE CORE SET 2016.

Models	G	T	F	C	RMSE↓	MAE↓	SD↓
DBGT-PLA-A	✓	×	×	×	2.561	2.099	1.787
DBGT-PLA-B	✓	×	×	✓	2.393	1.954	1.936
DBGT-PLA-C	✓	✓	×	×	1.529	1.317	1.456
DBGT-PLA-D	✓	✓	✓	×	1.439	1.216	1.379
DBGT-PLA-E(no-res.)	✓	✓	✓	×	1.447	1.258	1.411
DBGT-PLA-F(std.-T)	✓	✓	✓	×	1.471	1.243	1.459
DBGT-PLA	✓	✓	✓	✓	**1.231**	**1.096**	**1.225**

G: GC-GNN; T: Transformer (stable);
F: GRL Fusion; C: Composite Loss.
"No-res.": GRL without residual. "Std.-T": standard Transformer.

D. Ablation Study

To assess the contribution of key components within the DBGT-PLA architecture, we designed a series of ablation studies.

- **DBGT-PLA-A**
  Uses only the GC-GNN module with MSELoss. This baseline evaluates the performance of the pure GNN architecture without Transformer, fusion modules, or composite loss.
- **DBGT-PLA-B**
  Extends DBGT-PLA-A by introducing the composite loss as Eq. (4) while keeping the architecture unchanged. This variant isolates the contribution of the loss design to prediction accuracy.
- **DBGT-PLA-C**
  Combines GC-GNN and the proposed StableTransformerLayer but fuses features through direct concatenation. Local and global representations are simply stacked without alignment, gating, or feature selection, making this a baseline for evaluating global-context modeling independent of structured fusion. Loss remains MSELoss.
- **DBGT-PLA-D**
  Based on DBGT-PLA-C, this variant incorporates the full GRL fusion module (gating + residual path) while still using the StableTransformerLayer. It tests whether structured, gated-residual integration improves feature interaction beyond simple concatenation.
- **DBGT-PLA-E**
  Evaluates the impact of the residual path in GRL. It preserves the gating mechanism but removes the residual addition, using $f = g_w \cdot t$. This isolates the role of residual learning in stabilizing and enhancing the fusion of local and global features.

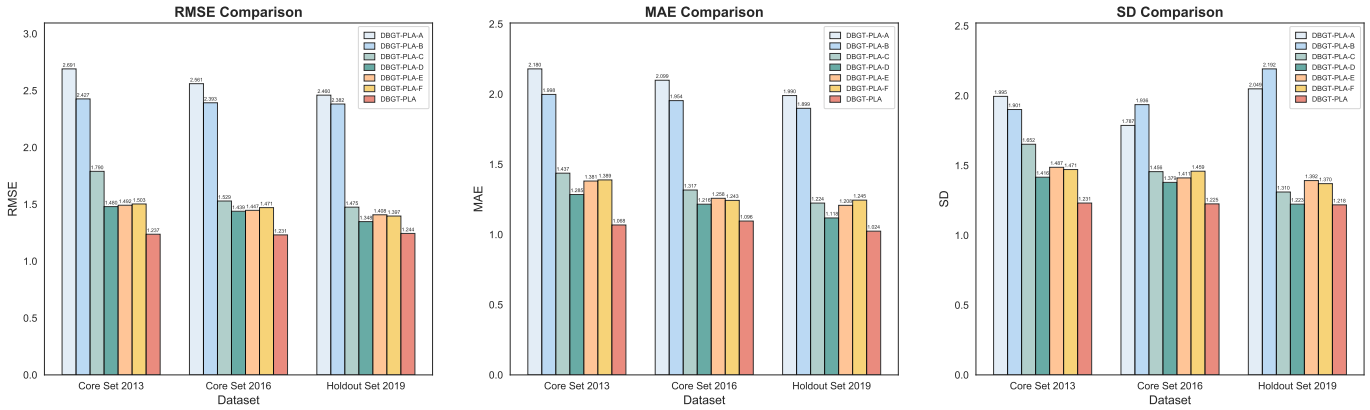

Fig. 5. Performance of different ablation variants of DBGT-PLA. Three panels correspond to the three metrics, namely RMSE, MAE, and SD.

TABLE VI
ABLATION STUDY RESULTS ON THE HOLDOUT SET 2019.

Models	G	T	F	C	RMSE↓	MAE↓	SD↓
DBGT-PLA-A	✓	×	×	×	2.460	1.990	2.049
DBGT-PLA-B	✓	×	×	✓	2.382	1.899	2.192
DBGT-PLA-C	✓	✓	×	×	1.475	1.224	1.310
DBGT-PLA-D	✓	✓	✓	×	1.348	1.118	1.223
DBGT-PLA-E(no-res.)	✓	✓	✓	×	1.408	1.208	1.392
DBGT-PLA-F(std.-T)	✓	✓	✓	×	1.397	1.245	1.370
DBGT-PLA	✓	✓	✓	✓	**1.244**	**1.024**	**1.218**

G: GC-GNN; T: Transformer (stable);
F: GRL Fusion; C: Composite Loss.
"No-res.": GRL without residual. "Std.-T": standard Transformer.

- **DBGT-PLA-F**
  Replaces the StableTransformerLayer with a standard Transformer while keeping the rest of the architecture unchanged. This ablation directly tests the contribution of numerical-stability mechanisms such as NaN filtering, value clipping, and learnable positional encodings.
- **DBGT-PLA**
  The full model combining GC-GNN, StableTransformer, GRL fusion, and the composite loss. This configuration evaluates the synergistic benefits of integrating all components.

Table IV–Table VI summarize the ablation results on the core set 2013, core set 2016, and the holdout set 2019. Overall, DBGT-PLA-A, which contains only the GC-GNN branch, performs the worst across all benchmarks, confirming that local structural information alone is insufficient for modeling complex protein–ligand interactions. Introducing the composite loss in DBGT-PLA-B does not yield improvements, indicating that advanced loss functions cannot compensate for limited representational capacity in the absence of stronger architectural components.

Incorporating the Transformer branch in DBGT-PLA-C leads to substantial performance gains on all datasets, demonstrating the importance of global contextual modeling beyond GNN-based local features. The GRL Fusion module in DBGT-PLA-D further improves accuracy in a stable and consistent manner, suggesting that structured feature integration enhances the coordination between local and global representations.

The ablation variants provide more detailed insights. Removing the residual path in the fusion module (DBGT-PLA-E) consistently degrades performance relative to DBGT-PLA-D, highlighting the value of residual learning in maintaining feature-scale balance and stabilizing cross-branch interactions. Replacing the StableTransformer with a standard Transformer (DBGT-PLA-F) also leads to noticeable performance drops across all benchmarks, demonstrating the necessity of the proposed stability-enhancing techniques.

Among all configurations, the full DBGT-PLA model achieves the best overall performance across the three datasets, indicating that each component contributes complementary benefits and together forms a robust and effective architecture for protein–ligand affinity prediction.

In summary, the integration of the Transformer module plays a pivotal role in improving model performance, while the GRL Fusion mechanism yields consistent gains and stabilizes the integration of local and global features. The residual path within GRL and the stability-enhanced Transformer are both empirically important, as removing either leads to measurable performance drops. The Composite Loss Function further refines optimization and improves the final accuracy. Overall, the complete DBGT-PLA framework achieves the best results across all datasets, demonstrating that the proposed components are complementary and jointly enhance both predictive performance and training robustness.

V. DISCUSSION

A. Potential technical impact

Our study introduces DBGT-PLA, a hybrid architecture that integrates graph neural networks with stability-enhanced Transformers, fused by a GRL module. This design enables the model to capture both local chemical interactions and global topological dependencies, which are critical for protein–ligand affinity prediction. Compared to existing baselines, the proposed model consistently achieves lower RMSE and higher correlation scores across multiple PDBbind benchmarks. More importantly, the ShapG-based interpretability analysis demonstrates that the model's predictions are not black-box outputs but are aligned with chemically meaningful substructures. These results suggest that SGFormer not only provides performance improvements but also enhances the

trustworthiness and transparency of deep learning in drug discovery workflows.

B. Potential Clinical Impact

The accurate and interpretable prediction of binding affinities has significant implications in structure-based drug design. By pinpointing substructures with high Shapley importance, our framework can guide medicinal chemists to prioritize modifications that improve binding efficiency. For example, functional groups highlighted in 5I13 and 6C0U could serve as rational starting points for ligand optimization, reducing the trial-and-error process in lead compound refinement. Furthermore, the ability to generalize across diverse benchmarks indicates potential for practical deployment in virtual screening pipelines, where reliability and interpretability are key for translational applications.

C. Future work

Despite its promising results, this work has several limitations. First, the current framework primarily focuses on static protein–ligand complexes, without explicitly modeling protein flexibility or induced fit effects. Future work could incorporate advanced graph learning techniques, such as the approximate generative Bayesian learning from LCAAG [48] to handle dynamic graph structures derived from molecular dynamics (MD) simulations, and the multiview fusion approach from FMvPCI [49] to integrate multiple conformational states (e.g., from MD trajectories) for capturing structural dynamics. This integration would enable biophysically consistent modeling of flexibility and induced fit, enhancing predictive accuracy in drug screening. Second, although ShapG provides valuable interpretability at the substructure level, extending the analysis to protein–ligand interface interactions (e.g., hydrogen bonds, salt bridges, π–π stacking) would provide more comprehensive biological insights. Finally, while this study focused on benchmark datasets, further validation on clinically relevant datasets is necessary to confirm the generalizability of the proposed framework.

VI. Conclusion

In this work, we proposed DBGT-PLA, a dual-branch fusion model integrating Graph Neural Networks and a stability-enhanced Transformer for interpretable protein–ligand binding affinity prediction. The model jointly captures local atomic interactions and long-range contextual dependencies via a gated residual fusion mechanism, while the composite loss function balances regression accuracy and correlation consistency. Extensive experiments on PDBbind benchmarks demonstrate that DBGT-PLA consistently outperforms state-of-the-art methods across multiple metrics.

Beyond predictive performance, the ShapG-based explainability analysis reveals chemically meaningful insights—such as the critical role of hydroxyl, aromatic, amide-pyrrolidine linker and indazole scaffold in affinity determination—highlighting that the model aligns well with established biochemical principles. This interpretable perspective bridges deep learning and medicinal chemistry, offering rational guidance for ligand optimization.

Overall, DBGT-PLA may act as a promising computational model for the accurate and efficient prediction of protein-ligand affinities.

VII. References

[1] L. J. Clark, G. Bu, B. L. Nannenga, and T. Gonen, "Microed for the study of protein–ligand interactions and the potential for drug discovery," *Nature Reviews Chemistry*, vol. 5, no. 12, pp. 853–858, 2021.

[2] J. M. Stokes, K. Yang, K. Swanson, W. Jin, A. Cubillos-Ruiz, N. M. Donghia, C. R. MacNair, S. French, L. A. Carfrae, Z. Bloom-Ackermann *et al.*, "A deep learning approach to antibiotic discovery," *Cell*, vol. 180, no. 4, pp. 688–702, 2020.

[3] M. K. Transtrum, L. D. Hansen, and C. Quinn, "Enzyme kinetics determined by single-injection isothermal titration calorimetry," *Methods*, vol. 76, pp. 194–200, 2015.

[4] A. Olaru, C. Bala, N. Jaffrezic-Renault, and H. Y. Aboul-Enein, "Surface plasmon resonance (spr) biosensors in pharmaceutical analysis," *Critical reviews in analytical chemistry*, vol. 45, no. 2, pp. 97–105, 2015.

[5] P. Lindström and O. Wager, "Igg autoantibody to human serum albumin studied by the elisa-technique," *Scandinavian journal of immunology*, vol. 7, no. 5, pp. 419–425, 1978.

[6] D. D. Wang, L. Ou-Yang, H. Xie, M. Zhu, and H. Yan, "Predicting the impacts of mutations on protein-ligand binding affinity based on molecular dynamics simulations and machine learning methods," *Computational and structural biotechnology journal*, vol. 18, pp. 439–454, 2020.

[7] D. D. Wang, M. Zhu, and H. Yan, "Computationally predicting binding affinity in protein–ligand complexes: free energy-based simulations and machine learning-based scoring functions," *Briefings in bioinformatics*, vol. 22, no. 3, 2021.

[8] S. Song, X. Chen, Y. Zhang, Z. Tang, and Y. Todo, "Protein–ligand docking using differential evolution with an adaptive mechanism," *Knowledge-Based Systems*, vol. 231, p. 107433, 2021.

[9] Q. Wu, Z. Peng, Y. Zhang, and J. Yang, "Coach-d: improved protein–ligand binding sites prediction with refined ligand-binding poses through molecular docking," *Nucleic acids research*, vol. 46, no. W1, pp. W438–W442, 2018.

[10] M. Karimi, D. Wu, Z. Wang, and Y. Shen, "Deepaffinity: interpretable deep learning of compound–protein affinity through unified recurrent and convolutional neural networks," *Bioinformatics*, vol. 35, no. 18, pp. 3329–3338, 2019.

[11] X. Liu, H. Feng, J. Wu, and K. Xia, "Dowker complex based machine learning (dcml) models for protein-ligand binding affinity prediction," *PLoS computational biology*, vol. 18, no. 4, p. e1009943, 2022.

[12] 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.

[13] D. D. Wang and M.-T. Chan, "Protein-ligand binding affinity prediction based on profiles of intermolecular contacts," *Computational and structural biotechnology journal*, vol. 20, pp. 1088–1096, 2022.

[14] P. A. Shar, W. Tao, S. Gao, C. Huang, B. Li, W. Zhang, M. Shahen, C. Zheng, Y. Bai, and Y. Wang, "Pred-binding: large-scale protein–ligand binding affinity prediction," *Journal of enzyme inhibition and medicinal chemistry*, vol. 31, no. 6, pp. 1443–1450, 2016.

[15] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *nature*, vol. 521, no. 7553, pp. 436–444, 2015.

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

[17] K. Wang, R. Zhou, Y. Li, and M. Li, "Deepdtaf: a deep learning method to predict protein–ligand binding affinity," *Briefings in Bioinformatics*, vol. 22, no. 5, p. bbab072, 2021.

[18] H. Wang, J. Tang, Y. Ding, and F. Guo, "Exploring associations of non-coding rnas in human diseases via three-matrix factorization with hypergraph-regular terms on center kernel alignment," *Briefings in Bioinformatics*, vol. 22, no. 5, p. bbaa409, 2021.

[19] 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.

[20] D. Jones, H. Kim, X. Zhang, A. Zemla, G. Stevenson, W. D. Bennett, D. Kirshner, S. E. Wong, F. C. Lightstone, and J. E. Allen, "Improved protein–ligand binding affinity prediction with structure-based deep fusion inference," *Journal of chemical information and modeling*, vol. 61, no. 4, pp. 1583–1592, 2021.

[21] D. Jiang, C.-Y. Hsieh, Z. Wu, Y. Kang, J. Wang, E. Wang, B. Liao, C. Shen, L. Xu, J. Wu *et al.*, "Interactiongraphnet: a novel and efficient deep graph representation learning framework for accurate protein–ligand interaction predictions," *Journal of medicinal chemistry*, vol. 64, no. 24, pp. 18 209–18 232, 2021.

[22] S. Chen, H. Yi, Z. You, X. Shang, Y.-A. Huang, L. Wang, and Z. Wang, "Local–global structure-aware geometric equivariant graph representation learning for predicting protein–ligand binding affinity," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 36, no. 8, pp. 15 181–15 193, 2025.

[23] G. Wang, H. Zhang, M. Shao, S. Sun, and C. Cao, "Mmpd-dta: Integrating multi-modal deep learning with pocket-drug graphs for drug-target binding affinity prediction," *Journal of Chemical Information and Modeling*, vol. 65, no. 3, pp. 1615–1630, 2025.

[24] G. Wang, H. Zhang, M. Shao, Y. Feng, C. Cao, and X. Hu, "Deeptgin: a novel hybrid multimodal approach using transformers and graph isomorphism networks for protein-ligand binding affinity prediction," *Journal of Cheminformatics*, vol. 16, no. 1, p. 147, 2024.

[25] D. Li, Y. Yang, Z. Cui, H. Yin, P. Hu, and L. Hu, "Llm-ddi: Leveraging large language models for drug-drug interaction prediction on biomedical knowledge graph," *IEEE Journal of Biomedical and Health Informatics*, 2025.

[26] X. Su, P. Hu, D. Li, B. Zhao, Z. Niu, T. Herget, P. S. Yu, and L. Hu, "Interpretable identification of cancer genes across biological networks via transformer-powered graph representation learning," *Nature biomedical engineering*, vol. 9, no. 3, pp. 371–389, 2025.

[27] Z. Yang, W. Zhong, Q. Lv, T. Dong, G. Chen, and C. Y.-C. Chen, "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*, vol. 46, no. 12, pp. 8191–8208, 2024.

[28] R. Dwivedi, D. Dave, H. Naik, S. Singhal, R. Omer, P. Patel, B. Qian, Z. Wen, T. Shah, G. Morgan *et al.*, "Explainable ai (xai): Core ideas, techniques, and solutions," *ACM computing surveys*, vol. 55, no. 9, pp. 1–33, 2023.

[29] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," *Advances in neural information processing systems*, vol. 30, 2017.

[30] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly *et al.*, "An image is worth 16x16 words: Transformers for image recognition at scale," *arXiv preprint arXiv:2010.11929*, 2020.

[31] A. Rives, J. Meier, T. Sercu, S. Goyal, Z. Lin, J. Liu, D. Guo, M. Ott, C. L. Zitnick, J. Ma *et al.*, "Biological structure and function emerge from scaling unsupervised learning to 250 million protein sequences," *Proceedings of the National Academy of Sciences*, vol. 118, no. 15, p. e2016239118, 2021.

[32] B. Fabian, T. Edlich, H. Gaspar, M. Segler, J. Meyers, M. Fiscato, and M. Ahmed, "Molecular representation learning with language models and domain-relevant auxiliary tasks," *arXiv preprint arXiv:2011.13230*, 2020.

[33] R. Murphy, B. Srinivasan, V. Rao, and B. Ribeiro, "Relational pooling for graph representations," in *International Conference on Machine Learning*. PMLR, 2019, pp. 4663–4673.

[34] Y. Seo, A. Loukas, and N. Perraudin, "Discriminative structural graph classification," *arXiv preprint arXiv:1905.13422*, 2019.

[35] M. Chatzianastasis, J. Lutzeyer, G. Dasoulas, and M. Vazirgiannis, "Graph ordering attention networks," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 37, no. 6, 2023, pp. 7006–7014.

[36] C. Morris, G. Rattan, and P. Mutzel, "Weisfeiler and leman go sparse: Towards scalable higher-order graph embeddings," *Advances in Neural Information Processing Systems*, vol. 33, pp. 21 824–21 840, 2020.

[37] G. Nikolentzos, G. Dasoulas, and M. Vazirgiannis, "K-hop graph neural networks," *Neural Networks*, vol. 130, pp. 195–205, 2020.

[38] C. Joo, H. Park, J. Lim, H. Cho, and J. Kim, "Machine learning-based heat deflection temperature prediction and effect analysis in polypropylene composites using catboost and shapley additive explanations," *Engineering Applications of Artificial Intelligence*, vol. 126, p. 106873, 2023.

[39] D. Lai, C. Demartino, and Y. Xiao, "Probabilistic machine leaning models for predicting the maximum displacements of concrete-filled steel tubular columns subjected to lateral impact loading," *Engineering Applications of Artificial Intelligence*, vol. 135, p. 108704, 2024.

[40] M. T. Kashifi, "Robust spatiotemporal crash risk prediction with gated recurrent convolution network and interpretable insights from shapley additive explanations," *Engineering Applications of Artificial Intelligence*, vol. 127, p. 107379, 2024.

[41] 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.

[42] C. Morris, M. Ritzert, M. Fey, W. L. Hamilton, J. E. Lenssen, G. Rattan, and M. Grohe, "Weisfeiler and leman go neural: Higher-order graph neural networks," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 33, no. 01, 2019, pp. 4602–4609.

[43] C. Zhao, J. Liu, and E. Parilina, "Shapg: new feature importance method based on the shapley value," *Engineering Applications of Artificial Intelligence*, vol. 148, p. 110409, 2025.

[44] Z. Yang, W. Zhong, Q. Lv, T. Dong, and C. Yu-Chian Chen, "Geometric interaction graph neural network for predicting protein–ligand binding affinities from 3d structures (gign)," *The journal of physical chemistry letters*, vol. 14, no. 8, pp. 2020–2033, 2023.

[45] Z. Jin, T. Wu, T. Chen, D. Pan, X. Wang, J. Xie, L. Quan, and Q. Lyu, "Capla: improved prediction of protein–ligand binding affinity by a deep learning approach based on a cross-attention mechanism," *Bioinformatics*, vol. 39, no. 2, p. btad049, 2023.

[46] X. Chen, J. Huang, T. Shen, H. Zhang, L. Xu, M. Yang, X. Xie, Y. Yan, and J. Yan, "Deattentiondta: protein–ligand binding affinity prediction based on dynamic embedding and self-attention," *Bioinformatics*, vol. 40, no. 6, p. btae319, 2024.

[47] S. Dandibhotla, M. Samudrala, A. Kaneriya, and S. Dakshanamurthy, "Gnnseq: A sequence-based graph neural network for predicting protein–ligand binding affinity," *Pharmaceuticals*, vol. 18, no. 3, p. 329, 2025.

[48] Y. Yang, L. Hu, G. Li, D. Li, P. Hu, and X. Luo, "Link-based attributed graph clustering via approximate generative bayesian learning," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 2025.

[49] ——, "Fmvpci: A multiview fusion neural network for identifying protein complex via fuzzy clustering," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 55, no. 9, pp. 6189–6202, 2025.