# LINKER: Learning Interactions Between FGs and Residues With Chemical Knowledge-Enhanced Reasoning and Explainability

**Phuc Pham***
Department of Computer Science
The University of Alabama at Birmingham
Alabama, United States
phpham@uab.edu

**Viet Thanh Duy Nguyen**
Department of Computer Science
The University of Alabama at Birmingham
Alabama, United States
dvnguye2@uab.edu

**Truong-Son Hy**†
Department of Computer Science
The University of Alabama at Birmingham
Alabama, United States
thy@uab.edu

## Abstract

Accurate mapping of protein-ligand interactions is key to understanding molecular recognition and guiding drug design. Existing deep learning methods typically depend on 3D structures or distance-based labels, limiting their scope and interpretability. We present LINKER, the first sequence-based model that predicts residue-functional group (FG) interactions in biologically defined types directly from protein sequences and ligand SMILES. Trained with structure-supervised attention using FG-derived labels from experimental complexes, LINKER focuses on chemically meaningful substructures and predicts interaction types such as hydrogen bonds and $\pi$-stacking. Crucially, it requires only sequence input at inference, enabling scalable application without structural data. Experiments on the LP-PDBBind benchmark show that LINKER achieves accurate, biochemically interpretable interaction maps aligned with ground-truth annotations.

## 1 Introduction

Protein–ligand interactions are central to chemical biology and pharmacology. Identifying which ligand functional groups (FGs) engage with specific protein residues is crucial for understanding molecular recognition, structure–activity relationships, and drug design. Existing methods depend on 3D complex structures; when absent, researchers rely on docking (e.g., AutoDock Vina [25]) and post-processing tools like PLIP [24], which are computationally costly and impractical for large-scale screening.

At present, no computational framework is capable of directly predicting biologically defined residue–functional group interactions from sequence-level input. Affinity-based models typically produce auxiliary maps with limited accuracy, while alternative strategies are largely restricted to pocket identification or distance estimation, lacking interpretability at the functional group level.

*This work is done during the author's research internship at the University of Alabama at Birmingham under the guidance of Dr. Truong-Son Hy.

†Corresponding author

39th Conference on Neural Information Processing Systems (NeurIPS 2025) 2nd Workshop on Multi-modal Foundation Models and Large Language Models for Life Sciences.

We present LINKER, the first framework to predict residue–FG interaction types directly from protein sequences and ligand SMILES. Trained with structure-supervised attention on PLIP-extracted interaction labels, LINKER learns chemically meaningful substructures and predicts interpretable interaction types (e.g., hydrogen bonds, $\pi$-stacking, salt bridges). At inference, it requires only sequence input, enabling scalable, structure-free prediction.

## 2 Problem Formulation

We frame residue–FG interaction prediction as a multi-label classification task. Given a protein $\mathbf{T}$ with $R$ residues and a ligand $\mathbf{D}$ with $F$ FGs, the goal is to estimate, for each residue-FG pair, probabilities over seven biologically defined interaction types: hydrogen bonds, hydrophobic contacts, $\pi$-stacking, $\pi$-cation interactions, salt bridges, water bridges, and halogen bonds. Formally,

$$f_{LINKER} : (\mathbf{T}, \mathbf{D}) \ \rightarrow \ \mathbf{P} \in [0,1]^{R \times F \times 7},$$

where $\mathbf{P}_{r,f,k}$ is the likelihood that residue $r$ and FG $f$ form interaction type $k$. Unlike contact maps that capture only spatial proximity, LINKER directly predicts interaction types, yielding chemically grounded and interpretable interaction maps.

## 3 Method

We propose LINKER, a sequence-based framework for predicting biologically defined residue–FG interactions from protein sequences and ligand SMILES (Fig. 1). LINKER integrates two modality-specific branches, a cross-attention module, and a pairwise interaction predictor (details in Appendix A).

**Protein branch** The protein sequence is encoded using the ESM C [5] protein language model, which is optimized to capture biologically meaningful representations of proteins. Unlike the ESM-3 [7] generative models focused on controllable sequence generation, *ESM C* specializes in representation learning and supports multi-chain inputs, making it more versatile than ESM-2 [14] for modeling complex protein-ligand interactions without requiring structural data.

**Ligand branch** Inspired by the development of multiscale models for complex chemical systems [10], we adopt a higher-level molecular representation based on functional groups rather than individual atoms. This abstraction aligns more naturally with protein-ligand interactions, which are typically governed by chemically meaningful substructures in the ligand and complementary residues in the protein. Traditional fingerprints such as the Morgan fingerprint [18], also known as the extended-connectivity fingerprint (ECFP4) [22], do capture local substructures, but they represent atom-centered environments without global context. As a result, the generated fragments often correspond to partial motifs that are difficult to consolidate into chemically interpretable functional groups, and multiple instances of the same group within a molecule cannot be distinguished. The lack of positional and semantic interpretability limits their explanatory power for binding mechanisms, thereby motivating our use of explicit functional group representations, which provide both chemical interpretability and biological relevance.

To address these limitations, we introduce two complementary modules. The *Functional Group Parser (FGParser)* deterministically maps each atom in a ligand SMILES to a functional group and interpolates the resulting group features back onto atoms to produce an $N_{\text{atom}} \times F$ atom-group representation matrix. Built upon this mapping, the *Functional Group and Positional Embedding for Molecular Identification (FINGER-ID)* learns multiscale representations by integrating local atomic neighborhood information with learnable functional-group embeddings that capture intrinsic chemical properties, while augmenting each group embedding with positional encodings and graph-level context derived from a Graph Convolutional Network (GCN) [11]. Our hierarchical architecture preserves atom–group correspondence and integrates signals across multiple scales, yielding position-aware and chemically interpretable representations that improve binding and molecular-property prediction.

**Interaction context integration** To model both intra- and inter-molecular dependencies, we introduce the *Self & Cross Attention Transformer (SCAT)* module. SCAT first applies self-attention [26]

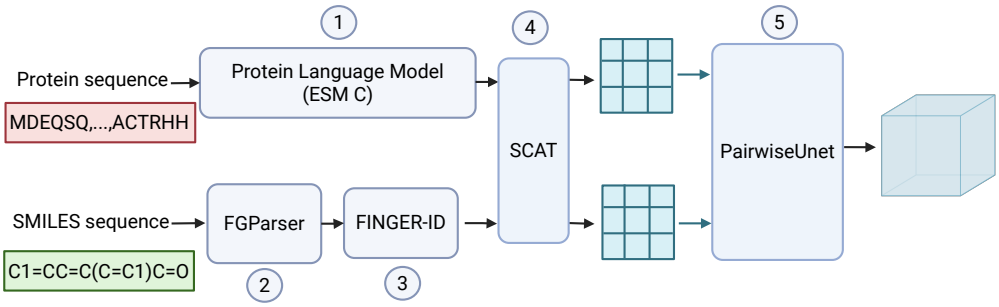



Figure 1: Overview of the LINKER framework for predicting biologically defined residue-functional group interaction types from protein sequences and ligand SMILES. The core components include: (1) Protein Language Model (ESM C), which encodes protein sequences into contextual residue embeddings; (2) Functional Group Parser (FGParser), which extracts functional groups from ligand SMILES and produces an atom–group representation matrix; (3) Functional Group and Positional Embedding for Molecular Identification (FINGER-ID), which generates context-aware functional group embeddings with positional and structural information; (4) Self & Cross Attention Transformer (SCAT), which integrates intra- and inter-molecular context between residues and functional groups; and (5) PairwiseUNet, which predicts interaction probability maps over all residue-functional group pairs and interaction types.

separately to the protein residue embeddings and ligand functional group embeddings to capture context within each modality. It then performs bidirectional cross-attention, where residue embeddings attend to functional group embeddings and vice versa, enabling the model to focus on residue–FG pairs with high interaction likelihood. The output is updated residue and functional group representations that are enriched with both local sequence/chemical context and cross-molecular interaction signals.

**Pairwise interaction prediction** The enriched protein and ligand representations produced by SCAT are integrated into a pairwise feature tensor, where each position corresponds to a residue–functional group pair. This tensor is then processed by *PairwiseUnet*, a customized architecture inspired by the 2D U-Net [23], designed to capture both local interaction motifs and global structural patterns within the interaction map. By casting interaction prediction as an image-like task, PairwiseUnet preserves fine-grained residue-group alignments while simultaneously modeling long-range dependencies, enabling the discovery of reusable biochemical motifs such as hydrogen bonds, hydrophobic contacts, and $\pi$-stacking. The network outputs a probability distribution over biologically defined interaction types for each residue-functional group pair, yielding binding maps that are both mechanistically meaningful and directly accessible from sequence-level input.

## 4 Experiments

We evaluate LINKER on held-out test data, comparing against representative baselines. Further details on dataset construction, baseline configurations, and evaluation metrics are provided in Appendix D.

### 4.1 Residue Interaction Prediction

We evaluate LINKER on binding-residue identification using binary labels from PLIP. For fair comparison with ArkDTA, which is residue-supervised, we convert residue–FG probabilities into residue-level indicators (see Appendix B.1 for the transformation procedure). Performance is assessed using PR and ROC curves over all residues, capturing robustness under class imbalance and overall discriminative power.

As shown in Figure 2, LINKER achieves a substantially higher area under the precision-recall curve (AP = 0.4073 vs. 0.2938 for ArkDTA, random baseline = 0.0243), indicating improved detection of true interaction residues under severe class imbalance. LINKER also attains a higher ROC AUC (0.9369 vs. 0.8688, random baseline = 0.5), confirming its superior overall discriminative ability. Together, these results highlight the effectiveness of our interaction supervision and LINKER's capacity to extract biologically meaningful binding signals from sequence-only inputs. Additional evidence of LINKER's advantage over ArkDTA is provided in Appendix B.2, where we evaluate both models under soft-label supervision.

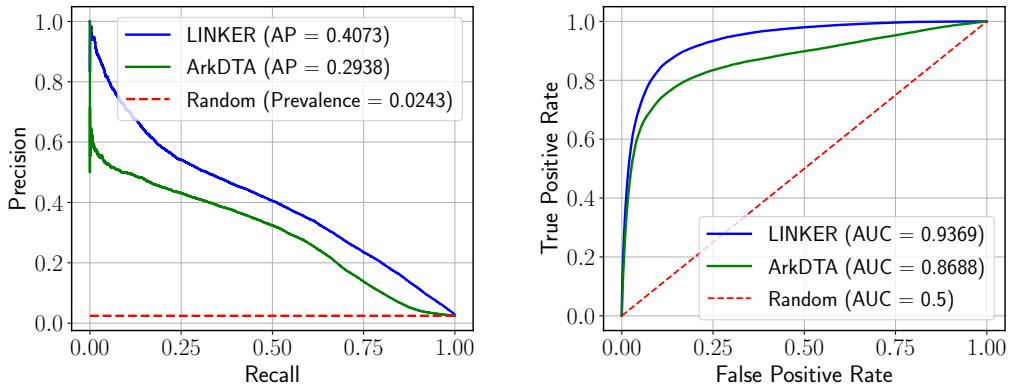



Figure 2: Residue-level interaction prediction. LINKER consistently surpasses ArkDTA in both PR *(left)* and ROC *(right)*. The PR analysis emphasizes robustness under class imbalance, while the ROC highlights stronger overall discrimination.

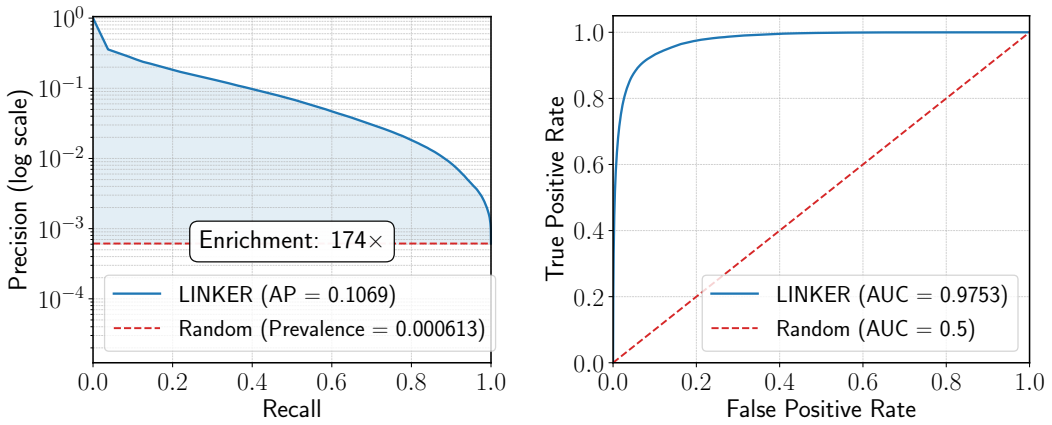
### 4.2 Residue-FG Interaction Prediction

We assess LINKER on the novel task of predicting fine-grained residue-FG interactions, using PLIP-derived labels as ground truth.

**Quantitative Evaluation.** Figure 3 reports LINKER's performance on residue–functional-group prediction. The PR curve (Fig. 3a) is shown against a random baseline (prevalence = 0.000613). Model strength is summarized by fold enrichment—defined as precision relative to prevalence—and all numerical evaluations, including fold enrichment and average precision, are consistently performed on the linear scale. Because prevalence is vanishingly small and raw precision values cluster near zero, we plot precision on a logarithmic axis for visualization only. Under this setting, LINKER achieves up to 174× enrichment over random at low recall, demonstrating substantial recovery of true residue–functional-group interactions despite extreme class sparsity (i.e., the positive class is exceedingly rare relative to negatives).

The ROC curve in Figure 3b shows the trade-off between the true positive rate and false positive rate. LINKER attains an AUC of 0.9753, indicating excellent discrimination between interacting and non-interacting residue–functional group pairs. Together, the PR and ROC analyses confirm that LINKER reliably captures rare interaction signals despite severe class imbalance.

**Qualitative Evaluation.** To further assess the interpretability of LINKER's predictions, we visualize the predicted residue-functional group interaction maps across four representative protein-ligand complexes in Figure 4. For each example, we compare the predicted interaction probabilities with the corresponding PLIP-derived ground truth. LINKER consistently highlights spatially localized interaction regions that align well with the annotated contacts, with the confidence scores highest for residue-group pairs in close spatial proximity to the ground truth. This shows the model's ability to infer chemically meaningful residue-group associations from sequence-level input while calibrating its predictions according to structural plausibility. Notably, even in complexes with sparse ground-truth labels, LINKER maintains high specificity by suppressing spurious interactions, illustrating its robust generalization. These examples confirm that LINKER produces interpretable and biologically grounded interaction maps across diverse structural contexts.




(a) PR curve with enrichment over the prevalence baseline (dashed).

(b) ROC curve showing strong discrimination between interacting and non-interacting pairs.

Figure 3: Residue-FG interaction prediction. LINKER delivers markedly higher enrichment at low recall and strong overall discrimination compared to a random baseline.

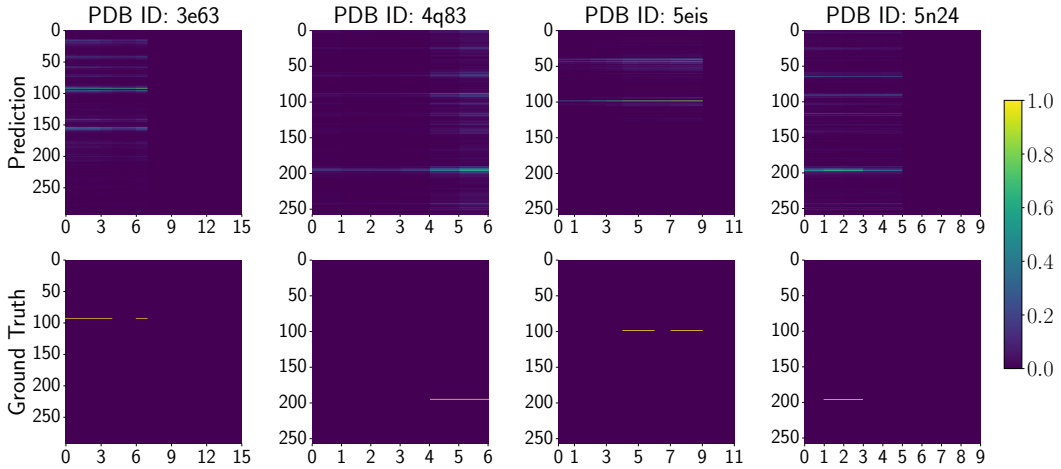



Figure 4: Comparison between LINKER predictions and PLIP ground truth for hydrophobic contacts across four protein structures. the x-axis shows FG indices derived from the atom-group matrix of the FGParser module A.2, and the y-axis corresponds to residue indices.

## 5 Conclusion

In this work, we introduced LINKER, a sequence-based model for predicting biologically defined residue-functional group interaction types directly from protein sequences and the ligand SMILES. By leveraging structure-supervised attention and motif extraction based on functional group, LINKER captures chemically meaningful substructures and predicts interaction types beyond simple spatial contacts. Operating entirely at the sequence level during inference, it enables large-scale application in scenarios lacking 3D structural data. Our experiments on the LP-PDBBind benchmark confirm that structure-informed supervision over functional group abstractions produces predictions that are highly consistent with ground-truth biochemical annotations, highlighting LINKER's potential as a practical and interpretable tool for molecular recognition and rational drug design.

In future work, we plan to evaluate LINKER on a broader range of protein-ligand interaction datasets to assess its generalization and robustness. We also aim to extend the framework to other types of biomolecular interactions, such as protein-protein, antibody-antigen, and protein-RNA binding, because the same core idea can be easily adapted to these tasks, broadening its applicability across diverse areas of structural biology and therapeutic design.

## References


[1] Rdkit: open-source cheminformatics http://www. rdkit. org.

[2] Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 2002.

[3] Pedro J Ballester and John BO Mitchell. A machine learning approach to predicting protein–ligand binding affinity with applications to molecular docking. *Bioinformatics*, 26(9):1169–1175, 2010.

[4] Jefferson Richard Dias. pyCheckmol: Application for detecting functional groups of a molecules. `https://github.com/jeffrichardchemistry/pyCheckmol`. [Accessed 13-08-2025].

[5] ESM Team. Esm cambrian: Revealing the mysteries of proteins with unsupervised learning, 2024.

[6] Mogan Gim, Junseok Choe, Seungheun Baek, Jueon Park, Chaeeun Lee, Minjae Ju, Sumin Lee, and Jaewoo Kang. Arkdta: attention regularization guided by non-covalent interactions for explainable drug–target binding affinity prediction. *Bioinformatics*, 39(Supplement_1):i448–i457, 2023.

[7] Thomas Hayes, Roshan Rao, Halil Akin, Nicholas J. Sofroniew, Deniz Oktay, Zeming Lin, Robert Verkuil, Vincent Q. Tran, Jonathan Deaton, Marius Wiggert, Rohil Badkundri, Irhum Shafkat, Jun Gong, Alexander Derry, Raul S. Molina, Neil Thomas, Yousuf A. Khan, Chetan Mishra, Carolyn Kim, Liam J. Bartie, Matthew Nemeth, Patrick D. Hsu, Tom Sercu, Salvatore Candido, and Alexander Rives. Simulating 500 million years of evolution with a language model. *Science*, 2025.

[8] Jiayue Hu, Yuhang Liu, Xiangxiang Zeng, Quan Zou, Ran Su, and Leyi Wei. Multi-modal deep representation learning accurately identifies and interprets drug-target interactions. *IEEE Journal of Biomedical and Health Informatics*, 2025.

[9] Dejun Jiang, Chang-Yu Hsieh, Zhenxing Wu, Yu Kang, Jike Wang, Ercheng Wang, Ben Liao, Chao Shen, Lei Xu, Jian Wu, et al. Interactiongraphnet: a novel and efficient deep graph representation learning framework for accurate protein–ligand interaction predictions. *Journal of medicinal chemistry*, 64(24):18209–18232, 2021.

[10] Martin Karplus. Development of multiscale models for complex chemical systems: From h+ h 2 to biomolecules (nobel lecture). *Angewandte Chemie International Edition*, 53(38), 2014.

[11] Thomas N. Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. In *International Conference on Learning Representations (ICLR)*, 2017.

[12] Jie Li, Xingyi Guan, Oufan Zhang, Kunyang Sun, Yingze Wang, Dorian Bagni, and Teresa Head-Gordon. Leak proof pdbbind: A reorganized dataset of protein-ligand complexes for more generalizable binding affinity prediction. *ArXiv*, pages arXiv–2308, 2024.

[13] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *Proceedings of the IEEE international conference on computer vision*, pages 2980–2988, 2017.

[14] Zeming Lin, Halil Akin, Roshan Rao, Brian Hie, Zhongkai Zhu, Wenting Lu, Nikita Smetanin, Allan dos Santos Costa, Maryam Fazel-Zarandi, Tom Sercu, Sal Candido, et al. Language models of protein sequences at the scale of evolution enable accurate structure prediction. *bioRxiv*, 2022.

[15] Zhihai Liu, Yan Li, Li Han, Jie Li, Jie Liu, Zhixiong Zhao, Wei Nie, Yuchen Liu, and Renxiao Wang. Pdb-wide collection of binding data: current status of the pdbbind database. *Bioinformatics*, 31(3):405–412, 2015.

[16] Andrew T McNutt, Abhinav K Adduri, Caleb N Ellington, Monica T Dayao, Eric P Xing, Hosein Mohimani, and David R Koes. Sprint enables interpretable and ultra-fast virtual screening against thousands of proteomes. *arXiv e-prints*, pages arXiv–2411, 2024.

[17] Nelson RC Monteiro, José L Oliveira, and Joel P Arrais. Dtitr: End-to-end drug–target binding affinity prediction with transformers. *Computers in Biology and Medicine*, 147:105772, 2022.

[18] Harry L Morgan. The generation of a unique machine description for chemical structures-a technique developed at chemical abstracts service. *Journal of chemical documentation*, 5(2):107–113, 1965.

[19] Viet Thanh Duy Nguyen and Truong Son Hy. Multimodal pretraining for unsupervised protein representation learning. *Biology Methods and Protocols*, 9(1):bpae043, 2024.

[20] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018.

[21] Hakime Öztürk, Arzucan Özgür, and Elif Ozkirimli. Deepdta: deep drug–target binding affinity prediction. *Bioinformatics*, 34(17):i821–i829, 2018.

[22] David Rogers and Mathew Hahn. Extended-connectivity fingerprints. *Journal of chemical information and modeling*, 50(5):742–754, 2010.

[23] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015.

[24] Sebastian Salentin, Sven Schreiber, V Joachim Haupt, Melissa F Adasme, and Michael Schroeder. Plip: fully automated protein–ligand interaction profiler. *Nucleic acids research*, 43(W1):W443–W447, 2015.

[25] Oleg Trott and Arthur J Olson. Autodock vina: improving the speed and accuracy of docking with a new scoring function, efficient optimization, and multithreading. *Journal of Computational Chemistry*, 31(2):455–461, 2010.

[26] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

[27] Tongzhou Wang and Phillip Isola. Understanding contrastive representation learning through alignment and uniformity on the hypersphere. In *International conference on machine learning*, pages 9929–9939. PMLR, 2020.

# A Detailed Methodology

## A.1 Protein Language Model (ESM C)

Let the input protein target be a FASTA sequence $\mathbf{T} = (t_1, \ldots, t_R),\ t_r \in \mathcal{A}$, where $\mathcal{A}$ denotes the amino acid alphabet and $R$ is the sequence length. The sequence is tokenized and passed through the ESM C encoder, which outputs residue-level embeddings

$$\mathbf{H}_p = f_\theta(\mathbf{T}) : \mathcal{A}^R \longrightarrow \mathbb{R}^{R\times D}, \qquad \mathbf{H}_p = \begin{bmatrix} \mathbf{h}_1^\top \\ \vdots \\ \mathbf{h}_R^\top \end{bmatrix} \in \mathbb{R}^{R\times D}.$$

The ESM C module provides contextualized residue embeddings, where each $\mathbf{h}_r$ captures both the local biochemical properties of residue $t_r$ and its long-range dependencies within the full protein sequence. These embeddings serve as the protein representation for downstream modules, enabling our framework to leverage rich structural and evolutionary information learned from large-scale protein corpora, without requiring explicit 3D structures. In particular, the residue-level features $\mathbf{H}_p$ are subsequently paired with ligand representations to facilitate accurate interaction and binding affinity prediction. In our setting, we use the ESM C (300M) variant with hidden size $D = 960$, resulting in an output matrix $\mathbf{H}_p \in \mathbb{R}^{R\times 960}$.

## A.2 Funtional Group Parser (FGParser)

The FGParser module decomposes a ligand into its constituent FGs and generates an atom-group representation matrix for downstream processing, as illustrated in Figure 5. The procedure comprises two stages: atom-group mapping and group-atom interpolation.

**Atom-group mapping.** Given a ligand SMILES sequence, denoted $\mathbf{D}$, FGParser first uses RDKit [1] to convert the SMILES into a Mol object, which encodes the atom and bond information in a structured format suitable for cheminformatics operations. The FGs are then identified using PyCheckmol [4], which applies a curated set of predefined chemical patterns to detect chemically significant substructures. Each detected FG is assigned a unique identifier and all atoms belonging to that group are tagged with the corresponding ID. This process produces an initial mapping of FGs to their member atoms, preserving explicit atom-group associations for downstream processing.

**Group-atom interpolation.** Some atoms may not belong to any canonical FG identified in the first stage. To ensure full coverage, these atoms are assigned to the nearest FG based on distance to functional-group atoms. Ties between equidistant groups are resolved by ascending group ID.

After this assignment step, we construct a binary matrix $\mathbf{M} \in \mathbb{R}^{N_{\text{atom}}\times F}$ defined as

$$\mathbf{M}_{i,j} = \begin{cases} 1 & \text{if atom } i \text{ belongs to or is assigned to FG } j, \\ 0 & \text{otherwise.} \end{cases}$$

This conversion preserves atom-level resolution while incorporating functional-group membership, enabling downstream modules to utilize both structural and chemical information.

## A.3 Funtional Group and Positional Embedding for Molecular Identification (FINGER-ID)

The FINGER-ID module generates hierarchical, context-aware embeddings for FGs identified by FGParser. Unlike conventional fingerprints such as Morgan fingerprints, which cannot capture positional information, distinguish multiple occurrences of the same group, or incorporate explicit structural context, FINGER-ID integrates signals across the atomic, meso, intermediate, and global scales. This design yields chemically interpretable representations suitable for diverse downstream tasks (Figure 6).

**Atomic-scale (Atomic Properties) Initialization.** We represent a molecule as a graph $G = (V, E)$ with $|V| = N$ atoms. Each atom $a \in V$ is associated with a feature vector $\mathbf{x}_a \in \mathbb{R}^{d_0}$ that encodes

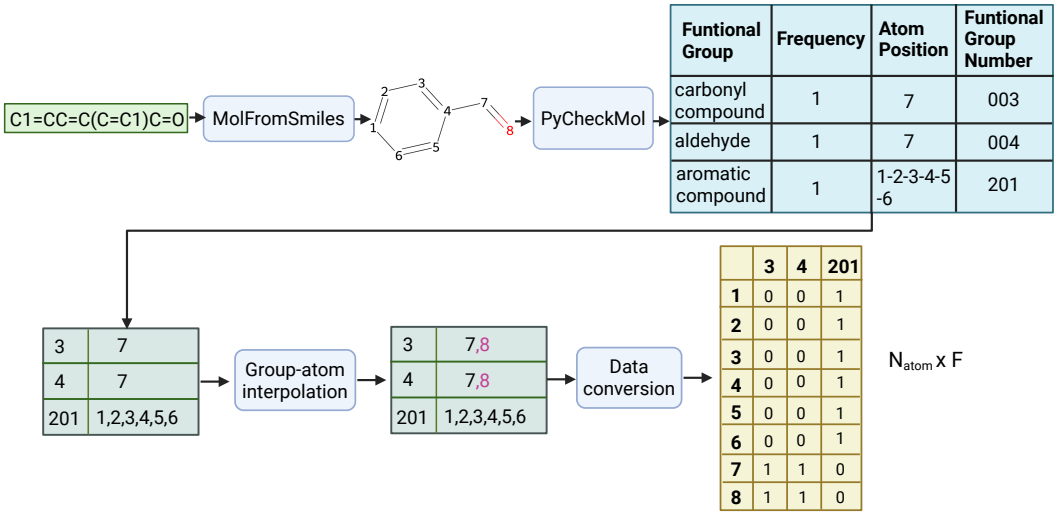



Figure 5: Overview of the FGParser module. Given a ligand SMILES, RDKit is used to construct a molecular graph, which is then analyzed by PyCheckmol to detect predefined FGs and generate initial atom-group mappings. A group-atom interpolation step ensures that all atoms are assigned to at least one FG. The final output is a binary atom-group matrix.

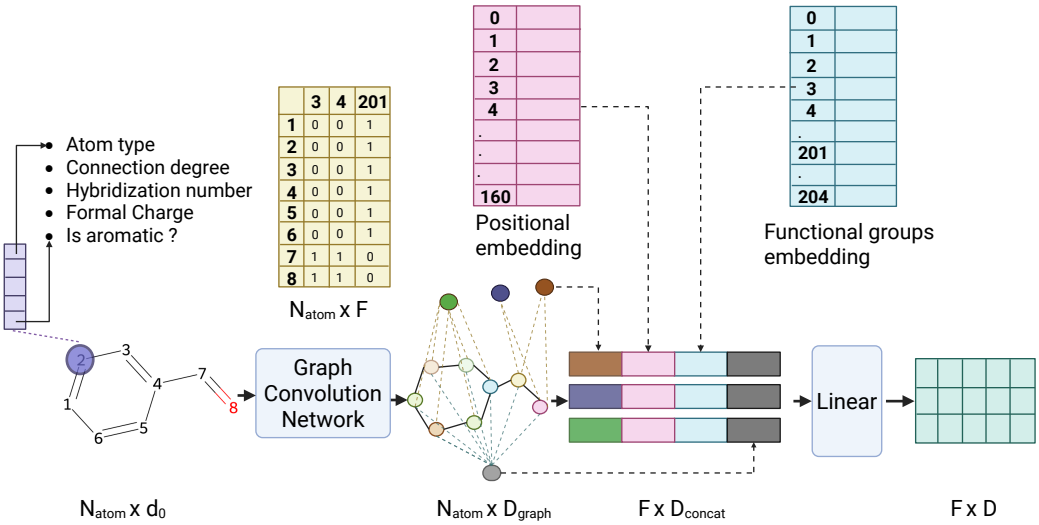



Figure 6: Atom-level features and chemical bonds define the molecular graph, which is encoded by a graph convolutional network (GCN). The resulting atom embeddings are aggregated into FG representations and enriched with group-type, positional, and global information. These enriched embeddings are then projected to form the final ligand representation. This hierarchical design captures structural signals from atomic, meso, intermediate, and global scales.

its atomic properties. In our implementation, we set $d_0 = 5$, corresponding to atom type, degree, hybridization, formal charge, and aromaticity. The node features are collected into a matrix

$$X = [\mathbf{x}_1^\top; \ldots; \mathbf{x}_N^\top] \in \mathbb{R}^{N \times d_0}.$$

The edge set $E$ is defined by the chemical bonds, capturing molecular connectivity.

**Meso-scale (Neighborhood) Encoding.** We encode the graph topology and node features with a graph convolutional network (GCN) [11] to obtain latent atom embeddings:

$$\mathbf{Z} = \mathrm{GCN}(G, X) \in \mathbb{R}^{N \times D_{\text{graph}}},$$

where the $a$-th row $\mathbf{z}_a$ represents the embedding of atom $a$. This stage captures meso-scale connectivity patterns within atomic neighborhoods based solely on adjacency information.

**Intermediate-scale (FG) Embedding.** Each FG $g$ is represented by aggregating the embeddings of its constituent atoms according to the atom-to-group mapping $\mathcal{M}(g) \subseteq V$ obtained from FGParser:

$$\mathbf{z}_g^{\text{inter}} = \mathrm{AGG}\{\mathbf{z}_a \mid a \in \mathcal{M}(g)\}, \quad g = 1, \ldots, F,$$

where AGG denotes mean pooling. The group-level representation is then concatenated with a learnable group-type embedding $\mathbf{e}_g^{\text{FG}}$ and a positional embedding $\mathbf{e}_g^{\text{pos}}$ to distinguish multiple occurrences of the same group.

**Global-scale (Molecular) Embedding.** A global embedding is derived by applying a READOUT operation over all atom embeddings:

$$\mathbf{z}^{\text{global}} = \mathrm{READOUT}(\mathbf{Z}).$$

Finally, each functional-group embedding is augmented with global molecular context:

$$\mathbf{h}_g = \left[\mathbf{z}_g^{\text{inter}} \parallel \mathbf{e}_g^{\text{FG}} \parallel \mathbf{e}_g^{\text{pos}} \parallel \mathbf{z}^{\text{global}}\right],$$

yielding a multiscale representation that integrates atomic features, meso-scale neighborhoods, functional-group information, and global molecular structure.

**Final Projection.** All enriched group embeddings are stacked and linearly projected to obtain the final ligand representation:

$$\mathbf{H} = [\mathbf{h}_1; \ldots; \mathbf{h}_F] \in \mathbb{R}^{F \times D_{\text{concat}}}, \quad \mathbf{H}_l = \mathbf{H}\mathbf{W} + \mathbf{b} \in \mathbb{R}^{F \times D}.$$

By integrating atomic, meso, intermediate, and global scale information, FINGER-ID produces position-aware, chemically interpretable embeddings that can differentiate identical FGs in distinct chemical environments, thereby enhancing downstream binding and molecular-property prediction.

### A.4 Self & Cross Attention Transformer (SCAT)

To capture both intra- and inter-molecular dependencies, we introduce the SCAT module, as illustrated in Figure 7. Let $\mathbf{H}_p \in \mathbb{R}^{R \times D}$ denote the protein residue embeddings obtained from ESM C, and $\mathbf{H}_l \in \mathbb{R}^{F \times D}$ denote the ligand FG embeddings produced by FINGER-ID.

**Self-attention.** We utilize the self-attention mechanism to further encode the residue and functional-group embeddings:

$$\mathbf{H}'_p = \mathrm{SA}_p(\mathbf{H}_p),$$
$$\mathbf{H}'_l = \mathrm{SA}_l(\mathbf{H}_l),$$

where $\mathrm{SA}_p$ and $\mathrm{SA}_l$ are transformer encoder blocks with self-attention applied within each modality.

**Cross-attention.** Furthermore, we use the cross-attention to encode the outputs from the self-attention:

$$\mathbf{H}''_p = \mathrm{CA}_{p \leftarrow l}(\mathbf{H}'_p, \mathbf{H}'_l),$$
$$\mathbf{H}''_l = \mathrm{CA}_{l \leftarrow p}(\mathbf{H}'_l, \mathbf{H}'_p),$$

where $\mathrm{CA}_{p \leftarrow l}$ attends from protein residues (queries) to ligand FGs (keys/values), and $\mathrm{CA}_{l \leftarrow p}$ performs the reverse.

The final outputs $\mathbf{H}''_p$ and $\mathbf{H}''_l$ are enriched with local context and cross-molecular interaction signals, providing chemically and biologically informed representations for downstream interaction prediction.

### A.5 PairwiseUNet

The PairwiseUNet module, as illustrated in Figure 8, takes the enriched protein and ligand representations from SCAT, $\mathbf{H}''_p \in \mathbb{R}^{R \times D}$ and $\mathbf{H}''_l \in \mathbb{R}^{F \times D}$, and transforms them into a structured pairwise feature tensor for interaction prediction.

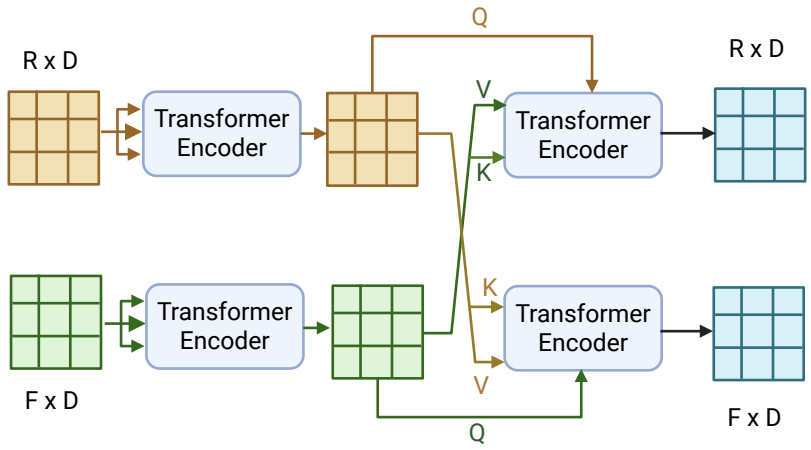



Figure 7: Overview of the Self & Cross Attention Transformer (SCAT) module, which integrates intra-molecular self-attention and inter-molecular cross-attention between protein residues and ligand FGs.

**Pairwise tensor construction** Protein embeddings are transmitted along the FG dimension to form $\mathbf{E}_p \in \mathbb{R}^{R\times F\times D}$, while ligand embeddings are transmitted along the residue dimension to form $\mathbf{E}_l \in \mathbb{R}^{R\times F\times D}$. These tensors are concatenated along the feature dimension, producing a joint representation:

$$\mathbf{Z} = \text{Concat}(\mathbf{E}_p, \mathbf{E}_l) \in \mathbb{R}^{R\times F\times 2D}.$$

**2D U-Net processing** The combined tensor $\mathbf{Z}$ is processed by a 2D U-Net, which models both local and long-range spatial dependencies across the residue–FG grid:

$$\mathbf{U} = \text{UNet}(\mathbf{Z}) \in \mathbb{R}^{R\times F\times D_{\text{unet}}}.$$

**Interaction type prediction** Finally, a stack of 2D convolutional layers [2] maps the pairwise feature tensor $\mathbf{U}$ to interaction-type probabilities:

$$\mathbf{P} = \text{Sigmoid}(\text{Conv2D}(\mathbf{U})) \in [0,1]^{R\times F\times K},$$

where $K = 7$ corresponds to biologically defined interaction types (e.g., hydrogen bonds, $\pi$–stacking, salt bridges). This produces an interpretable interaction map that can be directly derived from sequence-level input, with each entry $\mathbf{P}_{r,f,k}$ representing the independent probability of residue $r$ and FG $f$ participating in interaction type $k$.

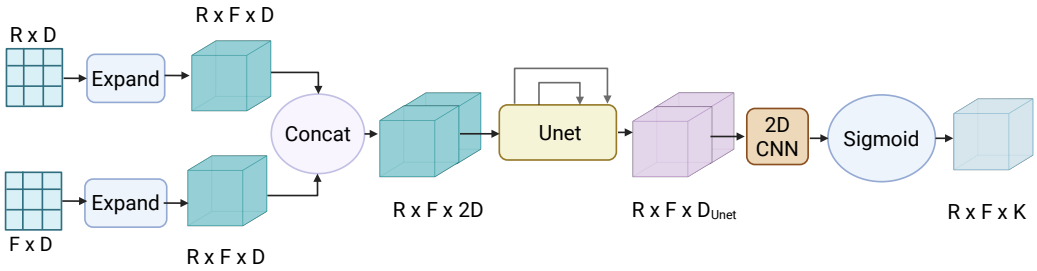



Figure 8: Overview of the PairwiseUNet module, which constructs residue–FG pairwise tensors and processes them with a U-Net–inspired 2D CNN architecture, followed by a sigmoid layer, to predict interaction probability maps.

# B Residue-Level Prediction

## B.1 Residue-Level Reduction of Interaction Probabilities

We evaluated the predictive performance of LINKER in identifying interaction residues, i.e., protein residues involved in ligand binding, using hard binary labels derived from PLIP annotations. To enable a fair comparison with ArkDTA, which is supervised only at the residue level, we aggregated our finer-grained residue-FG interaction labels into residue-level binary indicators, where a residue is marked as positive if it interacts with at least one ligand atom.

To enable this comparison, we derive residue-level interaction scores from LINKER's output tensor defined in Section 2, where the model predicts a tensor $\mathbf{P} \in [0,1]^{R \times F \times 7}$ representing interaction probabilities between $R$ protein residues and $F$ ligand FGs in seven biologically defined interaction types. We first aggregate over FGs by taking the maximum along the $F$-dimension:

$$\mathbf{U}_{r,k} = \max_{1 \le f \le F} \mathbf{P}_{r,f,k}, \quad \text{for } r = 1, \ldots, R \text{ and } k = 1, \ldots, 7,$$

resulting in a residue-wise interaction score matrix $\mathbf{U} \in [0,1]^{R \times 7}$. Next, we aggregate over interaction types:

$$\mathbf{y}_r = \max_{1 \le k \le 7} \mathbf{U}_{r,k}, \quad \text{for } r = 1, \ldots, R,$$

which produces a final prediction vector at the residue level $\mathbf{y} \in [0,1]^R$. This transformation produces a single interaction confidence score per residue, enabling a fair comparison with ArkDTA, which is supervised at the residue level. Importantly, this reduction preserves the fine-grained interaction information learned by LINKER while aligning it with the coarser evaluation setting.

## B.2 Residue Interaction Prediction with Soft Labels

Although PLIP-derived interaction labels provide reliable supervision, they are inherently binary and rely on strict geometric cutoffs, which can introduce boundary artifacts. In reality, ligand binding is often mediated by clusters of neighboring residues, where residues adjacent in sequence and spatially proximal may all contribute to recognition, even if only a subset is captured by PLIP's criteria. To account for this biological continuity and reduce sensitivity to cutoff-induced noise, we introduce Gaussian kernel smoothing along the residue dimension. Concretely, let us donote

$$y_{\text{hard}}[i] = \mathbb{I}\left[\sum_j \mathbf{Y}_{i,j} > 0\right]$$

as the binary residue-level label from PLIP. The smoothed label for the residue $i$ is then defined as

$$y_{\text{smooth}}[i] = \begin{cases} \max_{c \in H} \exp\left(-\dfrac{(i-c)^2}{2\sigma^2}\right), & H \neq \emptyset, \\ 0, & H = \emptyset, \end{cases} \quad H = \{\, c \mid y_{\text{hard}}[c] = 1 \,\}.$$

Here, PLIP-identified residues serve as anchor points with full confidence, while their immediate neighbors receive smoothly decaying support. Rather than altering the underlying ground truth, this smoothing acts as a biologically motivated regularization that reflects the clustered nature of binding sites. In doing so, it mitigates cutoff-induced artifacts and produces graded signals that enable a more flexible and faithful evaluation of interpretability.

Figure 9 illustrates the effect of smoothing with various standard deviations, showing how the binary interaction labels become progressively more diffuse as the smoothing parameter increases.

To evaluate model performance under this soft supervision regime, we computed the weighted precision of predicted residue-functional group interactions at different confidence thresholds, comparing LINKER and ArkDTA across smoothing levels. As shown in Figure 10, LINKER consistently outperforms ArkDTA on all smoothing scales and thresholds. Notably, LINKER achieves the highest weighted precision with a moderate smoothing parameter, indicating its ability not only to localize precise interaction sites but also to capture a broader biochemical context. In contrast, ArkDTA shows limited sensitivity to the smoothing factor, with relatively flat performance curves.

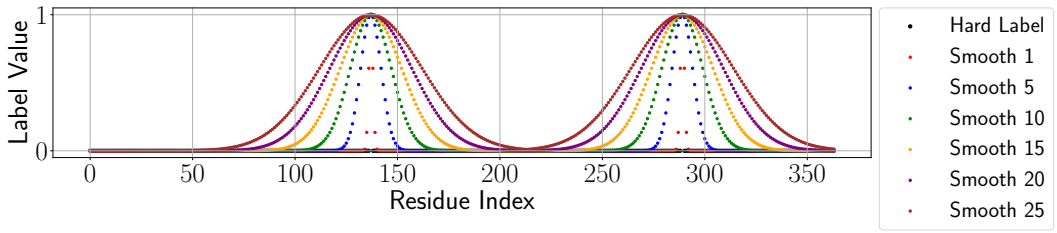



Figure 9: Visualization of residue-level interaction labels with Gaussian smoothing at different kernel widths. Wider kernels yield smoother label distributions, providing soft supervision signals around interacting residues.

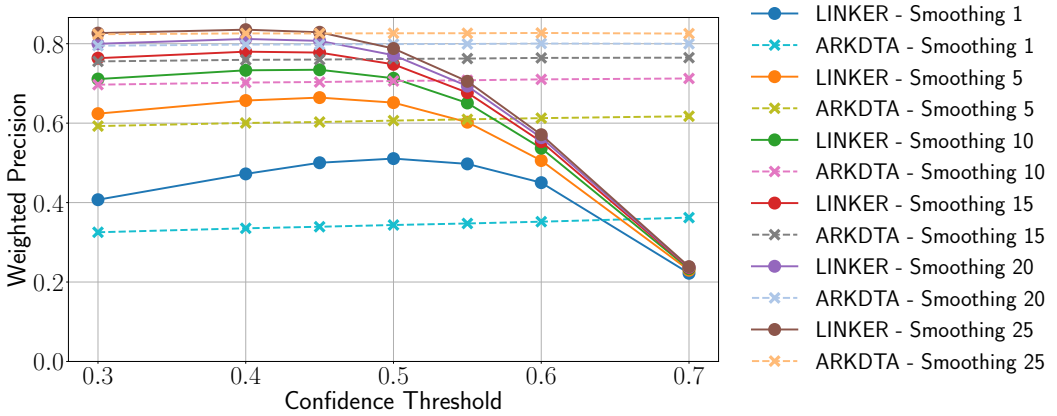



Figure 10: Weighted precision of LINKER and ArkDTA across various levels of label smoothing and confidence thresholds.

# C Binding Affinity Predictor

## C.1 Architecture

All weights of the modules in Figure 1 are frozen after training the interaction prediction module. Given target embeddings $\mathbf{H}_p \in \mathbb{R}^{R\times D}$ from the Protein Language Model (ESM-C) and ligand embeddings $\mathbf{H}_l \in \mathbb{R}^{F\times D}$ from FINGER-ID, along with interaction probabilities $\mathbf{P} \in \mathbb{R}^{R\times F\times K}$ from PairwiseUnet, the framework calculates a scalar prediction per protein-ligand complex (Figure 11).

**Contact Block.** This block fuses residue and FG information using predicted pairwise interaction probabilities. It first computes an edge-strength matrix that quantifies the importance of each residue-functional-group pair. From these strengths, it derives bidirectional attention coefficients and aggregates neighbor information into context vectors, which are then projected and added to the original embeddings to form enriched representations.

*Edge strength.* Edge strengths aggregate contributions from all interaction types into a single scalar per residue-functional-group pair:

$$S_{r,f} = \sum_{k=1}^{K} P_{r,f,k}\, w_k, \quad S = [S_{r,f}] \in \mathbb{R}^{R\times F},$$

where $w_k \in \mathbb{R}$ are learnable importance weights that reweight the predicted interaction-type probabilities.

*Bidirectional attention.* We convert edge strengths to normalized attention coefficients to capture the asymmetric relevance of each node to its neighbors:

$$\alpha_{r,f}^{p\to l} = \frac{\exp(S_{r,f})}{\sum_{f'=1}^{F}\exp(S_{r,f'})} \in \mathbb{R}, \quad \alpha_{r,f}^{l\to p} = \frac{\exp(S_{r,f})}{\sum_{r'=1}^{R}\exp(S_{r',f})} \in \mathbb{R}.$$

where $\alpha_{r,f}^{r\to f}$ measures the importance of the FG $f$ for residue $r$, and $\alpha_{r,f}^{f\to r}$ the opposite.

*Context aggregation & enriched embeddings.* Each residue aggregates a weighted sum of its functional-group neighbors to form a context vector, and each FG aggregates a weighted sum of its residue neighbors to form its context vector. These context vectors are then projected and added to the original embeddings to produce enriched representations:

$$\mathbf{c}_r^l = \sum_{f=1}^{F} \alpha_{r,f}^{p\to l}\, \mathbf{H}_l[f] \in \mathbb{R}^D, \quad \mathbf{c}_f^p = \sum_{r=1}^{R} \alpha_{r,f}^{l\to p}\, \mathbf{H}_p[r] \in \mathbb{R}^D,$$
$$\mathbf{H}_p'[r] = \mathbf{H}_p[r] + \mathrm{proj}_p(\mathbf{c}_r^l) \in \mathbb{R}^D, \quad \mathbf{H}_l'[f] = \mathbf{H}_l[f] + \mathrm{proj}_l(\mathbf{c}_f^p) \in \mathbb{R}^D,$$
$$\mathbf{H}_p' = [\mathbf{H}_p'[r]]_{r=1}^{R} \in \mathbb{R}^{R\times D}, \quad \mathbf{H}_l' = [\mathbf{H}_l'[f]]_{f=1}^{F} \in \mathbb{R}^{F\times D}.$$

Here, $\mathrm{proj}_r(\cdot)$ and $\mathrm{proj}_f(\cdot)$ denote (learned) linear projections that map context vectors back to the embedding space before residual addition.

**Fusion Block.** This block summarizes the enriched embeddings into compact pooled vectors and encodes a global signal for each interaction type. The pooled vectors are constructed so that nodes with stronger overall edge connections receive greater importance.

*Pooling weights and pooled embeddings.* We compute per-node importance scores by summing incident edge strengths, normalize them to obtain pooling weights, and form pooled embeddings as weighted sums of enriched node embeddings:

$$s_r^p = \sum_{f=1}^{F} S_{r,f} \in \mathbb{R}, \quad s_f^l = \sum_{r=1}^{R} S_{r,f} \in \mathbb{R},$$
$$\beta_r^p = \frac{s_r^p}{\sum_{r'} s_{r'}^p} \in \mathbb{R}, \quad \beta_f^l = \frac{s_f^l}{\sum_{f'} s_{f'}^l} \in \mathbb{R},$$
$$\mathbf{H}_p^{\text{pool}} = \sum_{r=1}^{R} \beta_r^p\, \mathbf{H}_p'[r] \in \mathbb{R}^D, \quad \mathbf{H}_l^{\text{pool}} = \sum_{f=1}^{F} \beta_f^l\, \mathbf{H}_l'[f] \in \mathbb{R}^D.$$

Intuitively, nodes with larger total edge strength contribute more to the pooled representation.

*Global interaction-type contribution.* A compact global vector records the overall contribution of each interaction type throughout the complex:

$$s_k = \sum_{r=1}^{R}\sum_{f=1}^{F} P_{r,f,k}\, w_k \in \mathbb{R}, \quad \mathbf{s} = [s_1, \ldots, s_K]^\top \in \mathbb{R}^K.$$

which captures the aggregate presence and importance of the interaction type $k$ in the current complex.

**Final representation and prediction.** We obtain the final feature vector by concatenating the pooled target embedding, the global interaction-type vector, and the pooled ligand embedding and then pass it to an MLP to produce the scalar binding affinity prediction.

$$\mathbf{h} = \left[\, \mathbf{H}_p^{\text{pool}},\ \mathbf{s},\ \mathbf{H}_l^{\text{pool}} \,\right] \in \mathbb{R}^{2D+K},$$
$$\hat{y} = \mathrm{MLP}(\mathbf{h}) \in \mathbb{R}.$$

### C.2 Transferability of LINKER Representations to Binding Affinity Prediction

To assess the generalization of our protein–ligand representations beyond interaction type prediction, we evaluated their performance on a downstream binding affinity regression task. This experiment validates whether LINKER captures biologically meaningful and transferable information about binding strength. Architectural specifications of the Binding Affinity Predictor are provided in Appendix C.1, and implementation details in Appendix D.

We perform this evaluation on the Leak-Proof PDBBind benchmark, a non-redundant, high-quality dataset designed to test model generalization under strict sequence and structural similarity constraints.

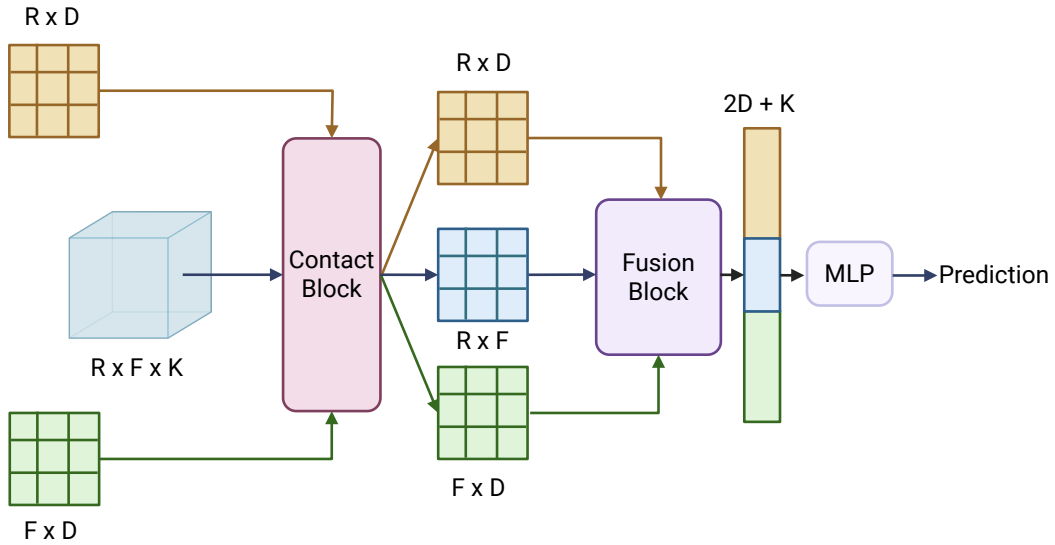



Figure 11: Overview of the binding affinity prediction framework. Target embeddings $\mathbf{H}_p$ from the protein language model and ligand embeddings $\mathbf{H}_l$ from the FINGER-ID module are combined with interaction probabilities $\mathbf{P}$ from the PairwiseUnet module. The **ContactBlock** encodes residue-FG interactions via edge strengths, bidirectional attention, and context aggregation, producing enriched embeddings. The **FusionBlock** pools these embeddings using learned importance weights and incorporates global interaction type contributions to form the final feature vector $\mathbf{h}$, which is fed into an MLP to predict the binding affinity.

To contextualize LINKER's performance, we compare it against a range of existing methods that also report results on this benchmark, spanning sequence-based, structure-based, and multimodal approaches. This enables a fair assessment of the transferability of LINKER's learned representations to a challenging real-world prediction task.

As shown in Table 1, LINKER achieves a test RMSE of 1.47, which is competitive with or better than several state-of-the-art methods specifically designed for the prediction of binding affinity. Notably, despite not being explicitly trained for this task, LINKER outperforms traditional docking-based approaches such as AutoDock Vina and learning-based baselines such as DeepDTA and IGN. Although its train and validation errors are slightly higher than those of MPRL and ArkDTA, both of which are affinity-focused models, LINKER achieves equivalent generalization performance on the test set, highlighting the transferability and biological relevance of its learned representations. This result confirms that LINKER captures fundamental interaction patterns that are predictive of binding strength, demonstrating strong potential for use in broader drug discovery pipelines.

Table 1: Comparison of RMSE on the Leak-Proof PDBBind benchmark for binding affinity prediction.

| Model | Train | Validation | Test |
|---|---|---|---|
| AutoDock Vina [25] | 2.42 | 2.29 | 2.56 |
| InteractionGraphNet (IGN) [9] | 1.65 | 2.00 | 2.16 |
| Random Forest (RF)-Score [3] | 0.68 | 2.14 | 2.10 |
| DeepDTA [21] | 1.41 | 2.07 | 2.29 |
| MPRL [19] | **0.48** | **1.47** | 1.55 |
| ArkDTA [6] | 1.18 | **1.47** | 1.48 |
| LINKER (Binding Affinity Predictor) | 1.38 | 1.53 | **1.47** |

# D Implementation Details

## D.1 Dataset

We evaluated LINKER on protein-ligand complexes from the PDBBind dataset [15], which provides high-quality structural data for a wide variety of binding interactions. Although PDBBind is commonly used for binding affinity prediction, in our work, it serves as a source of experimentally resolved 3D complexes from which we derive residue-FG interaction labels. Ground-truth labels are obtained by applying PLIP [24] to each complex after the ligands are decomposed into FGs.

To ensure a realistic and unbiased evaluation, we adopt the Leak-Proof PDBBind split [12], which eliminates structural redundancy between training and test sets by clustering protein-ligand complexes based on binding site similarity. This is particularly important for LINKER, as the supervision signal for attention is derived from 3D structural motifs. Preventing structural overlap ensures that LINKER cannot rely on memorized interaction patterns and must instead learn generalizable, interpretable mappings between sequence-level representations and biochemical interaction types.

## D.2 Baselines

Most prior models generate protein-ligand interaction maps as a secondary outcome of binding affinity prediction, typically using attention visualization or backpropagation rather than direct supervision [17, 16, 8]. These approaches often require 3D protein structures at inference time and may yield attention weights that do not reliably correspond to true biochemical interactions.

ArkDTA [6] stands out as the only sequence-based method that explicitly supervises cross-modal attention using residue-level interaction labels derived from 3D complex structures. By aligning attention weights between protein residues and ligand substructure tokens with non-covalent interaction (NCI) annotations, ArkDTA enables structure-free interpretability during inference, and thus serves as our primary baseline.

However, ArkDTA's supervision is limited to coarse residue-level labels and relies on Morgan fingerprint-based substructures, which lack chemical interpretability. In contrast, our method predicts fine-grained residue-FG interactions using chemically meaningful groups extracted by PyCheckMol. To ensure a fair comparison despite these differences, we evaluated both models on the shared task of predicting interaction residues under aligned supervision settings.

## D.3 Training Procedure

We trained our model on the LP-PDBBind dataset, which contains protein-ligand complexes with annotated 3D structures and experimentally measured binding affinities. Residue-FG interaction labels were derived using PLIP for geometric annotation and PyCheckMol for chemical FG decomposition.

**LINKER training.** To address the extreme class imbalance inherent in interaction annotations, we adopt the Focal Loss [13], which dynamically weights down-weights well-classified examples and emphasizes harder misclassified instances. The Focal Loss is defined as:

$$\mathcal{L}_{\text{focal}} = -\alpha(1 - p_t)^{\gamma} \log(p_t),$$

where $p_t$ is the predicted probability for the true class, $\alpha$ balances the contribution of positive and negative examples, and $\gamma$ controls the degree of focus on hard examples. In our experiments, we set $\alpha = 0.85$ and $\gamma = 1.0$.

**Binding Affinity Predictor training.** For downstream regression, we introduce an interaction-aware neural network (see Appendix C.1) that integrates residue embeddings, FG embeddings, and interaction probabilities produced by LINKER. To train the binding affinity predictor while regularizing the latent space, we optimize a combination of the Mean Squared Error (MSE) loss and the latent alignment loss:

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{MSE}} + \beta\, \mathcal{L}_{\text{latent}},$$

where $\beta$ is a hyperparameter controlling the contribution of the latent alignment term.

The MSE loss measures the difference between predicted and true binding affinities:

$$\mathcal{L}_{\text{MSE}} = \frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2,$$

where $y_i$ and $\hat{y}_i$ denote the ground truth and predicted binding affinities, respectively.

The latent alignment loss combines an InfoNCE term [20] and a uniformity term [27]:

$$\mathcal{L}_{\text{latent}} = \mathcal{L}_{\text{InfoNCE}} + \lambda\, \mathcal{L}_{\text{uniform}},$$

where $\lambda$ balances the contribution of the uniformity term.

The latent alignment loss encourages embeddings of samples with similar binding affinities to be close in the latent space while maintaining uniformity in the hypersphere. Let $\mathbf{h}$ denote the final representation defined in Appendix C.1, and let $\mathbf{h}_i$ be the representation of the $i$-th sample in a batch. We first normalize each sample embedding to obtain:

$$\mathbf{z}_i = \frac{\mathbf{h}_i}{\|\mathbf{h}_i\|_2} \in \mathbb{R}^{2D+K}.$$

which projects all embeddings onto the unit hypersphere. This normalization removes the influence of vector magnitude and ensures that similarity is determined solely by the angular distance, thereby preventing trivial solutions where embeddings collapse or scale arbitrarily.

The InfoNCE loss encourages embeddings of samples with similar binding affinities to be close to each other in the latent space. For each sample $i$, we define its positive sample $p(i)$ as the sample in the batch with the most similar binding score (excluding itself). The loss is then computed as:

$$\mathcal{L}_{\text{InfoNCE}} = -\frac{1}{B}\sum_{i=1}^{B}\log\frac{\exp\left(\mathbf{z}_i^\top \mathbf{z}_{p(i)}/\tau\right)}{\sum_{j=1}^{B}\exp\left(\mathbf{z}_i^\top \mathbf{z}_j/\tau\right)},$$

where $\tau$ is a temperature hyperparameter that controls the sharpness of the similarity distribution.

The Uniformity loss encourages latent vectors to spread on the hypersphere:

$$\mathcal{L}_{\text{uniform}} = \log\frac{1}{B^2}\sum_{i,j=1}^{B}\exp\Big(-2\|\mathbf{z}_i - \mathbf{z}_j\|_2^2\Big).$$

In our experiments, we set $\beta = 2$, $\lambda = 0.1$, and $\tau = 0.1$.

**Optimization Setup.** LINKER was trained for 30 epochs and the Binding Affinity Predictor for 80 epochs, both using the Adam optimizer with a learning rate of $2 \times 10^{-5}$ and batch sizes of 2 and 16, respectively. Validation performance was monitored after each epoch to assess generalization.

**Implementation Environment.** All experiments were implemented in PyTorch (v2.6.0) with CUDA 12.4 and run on an NVIDIA Tesla P100 GPU (16 GB VRAM).

### D.4 Evaluation Metric

We use the following evaluation metrics in our experiments:

**Precision-Recall Curve (PR Curve).** The PR curve evaluates the trade-off between precision and recall across varying thresholds:

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}, \qquad \text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}.$$

The area under the PR curve (PR AUC) summarizes model performance under class imbalance, where positive labels are sparse.

**Receiver Operating Characteristic Curve (ROC Curve).** The ROC curve plots the true positive rate (TPR) against the false positive rate (FPR):

$$\text{TPR} = \frac{\text{TP}}{\text{TP} + \text{FN}}, \qquad \text{FPR} = \frac{\text{FP}}{\text{FP} + \text{TN}}.$$

The area under the ROC curve (ROC AUC) reflects the model's ability to discriminate between positive and negative classes.

**Weighted Precision.** For soft labels, we compute weighted precision by thresholding the model's attention scores and weighting predictions by the smoothed label values:

$$\text{Weighted Precision} = \frac{\sum_{i=1}^{N} \hat{y}_i \cdot y_i}{\sum_{i=1}^{N} \hat{y}_i},$$

where $\hat{y}_i \in \{0, 1\}$ is the binary prediction and $y_i \in [0, 1]$ is the smoothed supervision label.

**Prevalence.** Prevalence is simply the proportion of positive samples in the dataset:

$$\text{Prevalence} = \frac{1}{N} \sum_{i=1}^{N} y_i,$$

which corresponds to the horizontal baseline of random predictions in the PR curve.

**Enrichment.** Enrichment measures the improvement of the precision of a model over the prevalence baseline:

$$\text{Enrichment} = \frac{\text{Precision}}{\text{Prevalence}}.$$

High enrichment values indicate that the model retrieves substantially more true positives than expected under random selection, which is particularly informative at low recall in highly imbalanced settings.

**Root Mean Squared Error (RMSE).** For regression tasks, we use the RMSE to quantify the deviation between predicted and ground-truth values:

$$\text{RMSE} = \sqrt{\frac{1}{N} \sum_{i=1}^{N} (\hat{y}_i - y_i)^2},$$

where $\hat{y}_i$ denotes the predicted binding affinity and $y_i$ is the corresponding ground-truth label. Lower RMSE values indicate better predictive accuracy.