Bioinformatics Advances, 2023, vbad070
https://doi.org/10.1093/bioadv/vbad070
Advance access publication 2 June 2023
Original Article

Structural bioinformatics

PIQLE: protein–protein interface quality estimation by deep graph learning of multimeric interaction geometries

Md Hossain Shuvo[1], **Mohimenul Karim**[1], **Rahmatullah Roche**[1] and **Debswapna Bhattacharya**[1,*]

[1]Department of Computer Science, Virginia Tech, Blacksburg, VA 24061, USA

*To whom correspondence should be addressed.

Associate Editor: Michael Gromiha

Abstract

Motivation: Accurate modeling of protein–protein interaction interface is essential for high-quality protein complex structure prediction. Existing approaches for estimating the quality of a predicted protein complex structural model utilize only the physicochemical properties or energetic contributions of the interacting atoms, ignoring evolutionarily information or inter-atomic multimeric geometries, including interaction distance and orientations.

Results: Here, we present PIQLE, a deep graph learning method for protein–protein interface quality estimation. PIQLE leverages multimeric interaction geometries and evolutionarily information along with sequence- and structure-derived features to estimate the quality of individual interactions between the interfacial residues using a multi-head graph attention network and then probabilistically combines the estimated quality for scoring the overall interface. Experimental results show that PIQLE consistently outperforms existing state-of-the-art methods including DProQA, TRScore, GNN-DOVE and DOVE on multiple independent test datasets across a wide range of evaluation metrics. Our ablation study and comparison with the self-assessment module of AlphaFold-Multimer repurposed for protein complex scoring reveal that the performance gains are connected to the effectiveness of the multi-head graph attention network in leveraging multimeric interaction geometries and evolutionary information along with other sequence- and structure-derived features adopted in PIQLE.

Availability and implementation: An open-source software implementation of PIQLE is freely available at https://github.com/Bhattacharya-Lab/PIQLE.

Contact: dbhattacharya@vt.edu

Supplementary information: Supplementary data are available at *Bioinformatics Advances* online.

1 Introduction

Protein–protein interactions are the actuators of numerous biological processes (Peng *et al.*, 2017). Despite the remarkable progress in predicting single-chain protein structures with a very high degree of accuracy (Baek *et al.*, 2021; Jumper *et al.*, 2021; Wallner, 2022), modeling the structures of protein complexes remains challenging (Bryant *et al.*, 2022; Evans *et al.*, 2022; Zahiri *et al.*, 2020). Traditional protein–protein docking approaches as well as recent deep learning-based protein complex structure prediction methods typically generate a number of candidate structural models and rank them based on estimated confidence scores to select the top-ranked model (Bryant *et al.*, 2022; Christoffer *et al.*, 2021; Lyskov and Gray, 2008; Pierce *et al.*, 2014). With the state-of-the-art protein structure prediction methods approaching near-experimental accuracy on single-chain predictions, accurately modeling the protein–protein interaction interfaces is the key to successfully predicting the structures of protein complexes. As such, high-fidelity estimation of the modeling quality of protein–protein interaction interface from a computationally predicted complex structure is critically important for characterizing protein–protein interactions (Cao and Shen, 2020; Vajda *et al.*, 2013).

Encouraging progress has been made in protein complex scoring and quality estimation. Physics-based approaches, such as ZRANK (Pierce and Weng, 2007), demonstrate effective scoring performance using the weighted sum of several energy terms including van der Waals force, hydrogen bonding, electrostatics, pair potentials and solvation. ZRANK2 (Pierce and Weng, 2008) further improves the scoring performance by optimizing certain energy terms used in ZRANK. In addition to physics-based approaches, state-of-the-art methods apply machine learning for the quality estimation of complex models. For example, TRScore (Guo *et al.*, 2022) estimates the quality of protein complex models by learning from a voxelized 3D grid representation of the protein–protein interface using a deep convolutional RepVGG architecture. DOVE (Wang *et al.*, 2020b) applies a 3D convolutional neural network (3DCNN) with voxelized representation of protein complexes while incorporating atomic interaction types and their energetic contributions. Additionally, it integrates knowledge-based statistical potentials GOAP (Zhou and Skolnick, 2011) and ITScore (Huang and Zou, 2008) to capture atomic interaction energies, demonstrating competitive scoring performance. Recently, representation learning with graph neural

Received: February 22, 2023. **Revised:** May 17, 2023. **Editorial Decision:** June 1, 2023. **Accepted:** June 1, 2023

networks (GNNs) (Zhou *et al.*, 2020) is gaining significant attention, leading to the development of several protein complex model quality estimation methods. For example, GNN-DOVE (Wang *et al.*, 2021) uses a graph attention network (GAT) (Veličković *et al.*, 2018) by embedding protein complex interfaces as graphs. DProQA (Chen *et al.*, 2023) uses a gated-graph transformer model for complex quality estimation.

Despite the progress, these methods do not consider two key factors that can significantly improve protein–protein interface quality estimation performance. First, the geometry of the interaction interface often carries key information about the spatial organization of the interacting partners and therefore provides a rich representation of a complex structure (Dai and Bailey-Kellogg, 2021; Ganea *et al.*, 2022), but none of the protein complex scoring methods incorporate multimeric interaction geometries, including the inter-atomic distance and orientations of the residues at the interaction interface. Second, while knowledge-based methods, such as InterEvScore (Andreani *et al.*, 2013), incorporate evolutionary information for scoring heteromeric protein complexes, the state-of-the-art machine learning-based approaches typically rely on the physicochemical properties or energetic contributions of the interacting atoms without considering the availability of evolutionarily information in the form of multiple sequence alignments (MSAs). That is, they ignore the effect of MSAs during scoring.

Here, we present a protein–protein interface quality estimation method called PIQLE by deep graph learning of multimeric interaction geometries. PIQLE formulates protein–protein interface quality estimation as a graph learning task by constructing a graph considering the residues at the interaction interface and estimates the interface quality by training a multi-head GAT using sequence- and structure-derived node features along with evolutionarily information and newly introduced edge features in the form of inter-atomic interaction distance and orientations capturing multimeric interaction geometries. Unlike the existing GNN-based methods operating on voxelized representation of the protein–protein interface to estimate the overall interface quality, PIQLE first estimates the quality of the individual interactions between the interfacial residues by edge-level error regression and then probabilistically combines the estimated quality of the interfacial residues for scoring the overall interface. Large-scale benchmarking on multiple widely used protein docking decoy sets demonstrates that PIQLE consistently attains better performance than existing complex model quality estimation methods in terms of various evaluation measures including hit rate, success rate, reproducibility of model-native similarity scores and distinguishability between acceptable and incorrect models. By conducting rigorous ablation study and comparison with the self-assessment module of AlphaFold-Multimer repurposed for protein complex scoring on an independent dataset, we directly verify that the improved performance of our method is connected to the effectiveness of the multi-head GAT in leveraging multimeric interaction geometries and evolutionary information along with the other sequence- and structure-derived features. PIQLE is freely available at https://github.com/Bhattacharya-Lab/PIQLE.

2 Materials and methods

Figure 1 illustrates our protein–protein interface quality estimation framework consisting of a graph representation of the interaction interface, featurization including multimeric interaction geometries and quality estimation of the individual interacting residues by edge-level error regression using multi-head GAT followed by probabilistic combination for the estimation of the overall interface quality.

2.1 Graph representation and featurization

We represent the protein–protein interface as a graph $G = (V, E)$, in which a node $v \in V$ represents an interface residue and an edge $e \in E$ represents an interacting interface residue pair. We consider an interface residue pair to be interacting if their C_β atoms (C_α for glycine) are within 10 Å (Marze *et al.*, 2018). With such a graph representation, we use a total of 17 node features and 27 edge features describing each interface residue and their interactions including sequence- and structure-based node features and multimeric interaction geometric edge features. We describe them below.

2.1.1 Node features

Residue encoding: We cluster 20 naturally occurring amino acids into 4 classes including polar, non-polar, positively charged and negatively charged (Supplementary Table S1) (Kumar *et al.*, 2018; Tavafoghi and Cerruti, 2016; Zhu *et al.*, 2016). For a given residue belonging to one of these classes, we perform one-hot encoding of the residue using five class bins with the last bin reserved for the non-standard amino acids belonging to none of the four preceding bins, leading to five features for each node in the interfacial graph (i.e. a binary vector of five entries).

Relative residue positioning: To capture the relative positional information for each residue, we extract one feature for each node in the interfacial graph corresponding to each of the amino acid residues in a sequence as follows:

$$\mathrm{relPos}(aa) = \frac{aa^n}{L},$$

where aa^n is the position of the nth residue in the sequence and L is the length of the sequence.

Secondary structure and solvent accessibility: We use DSSP (Kabsch and Sander, 1983) program to calculate the secondary structure and solvent accessibility from the structure. We transform eight-state secondary structures into three-state by grouping them into helices, strands and coils for each of the residues in the sequence (Supplementary Table S1). Additionally, we discretize the real-valued solvent accessibility into two states of buried and exposed using the solvent-accessible surface area for the corresponding residue (Supplementary Table S1). We then use the one-hot encoding of three-state secondary structures and two-state solvent accessibility, resulting in five features.

Local backbone geometry: We calculate phi (ϕ) and psi (ψ) backbone torsion angles from the structure to capture the local backbone geometry of each residue. We perform sinusoidal and cosine transformations of the angles (Li *et al.*, 2017), leading to four features.

Evolutionarily information: We compute the number of effective sequences (N_{eff}) from the MSAs of the individual monomer and concatenated MSA of the complex to account for the depth of the MSAs, thereby considering the availability of evolutionarily information. To generate MSAs from an individual monomeric sequence, we run HHblits (Remmert *et al.*, 2011) for three iterations with an *E*-value inclusion

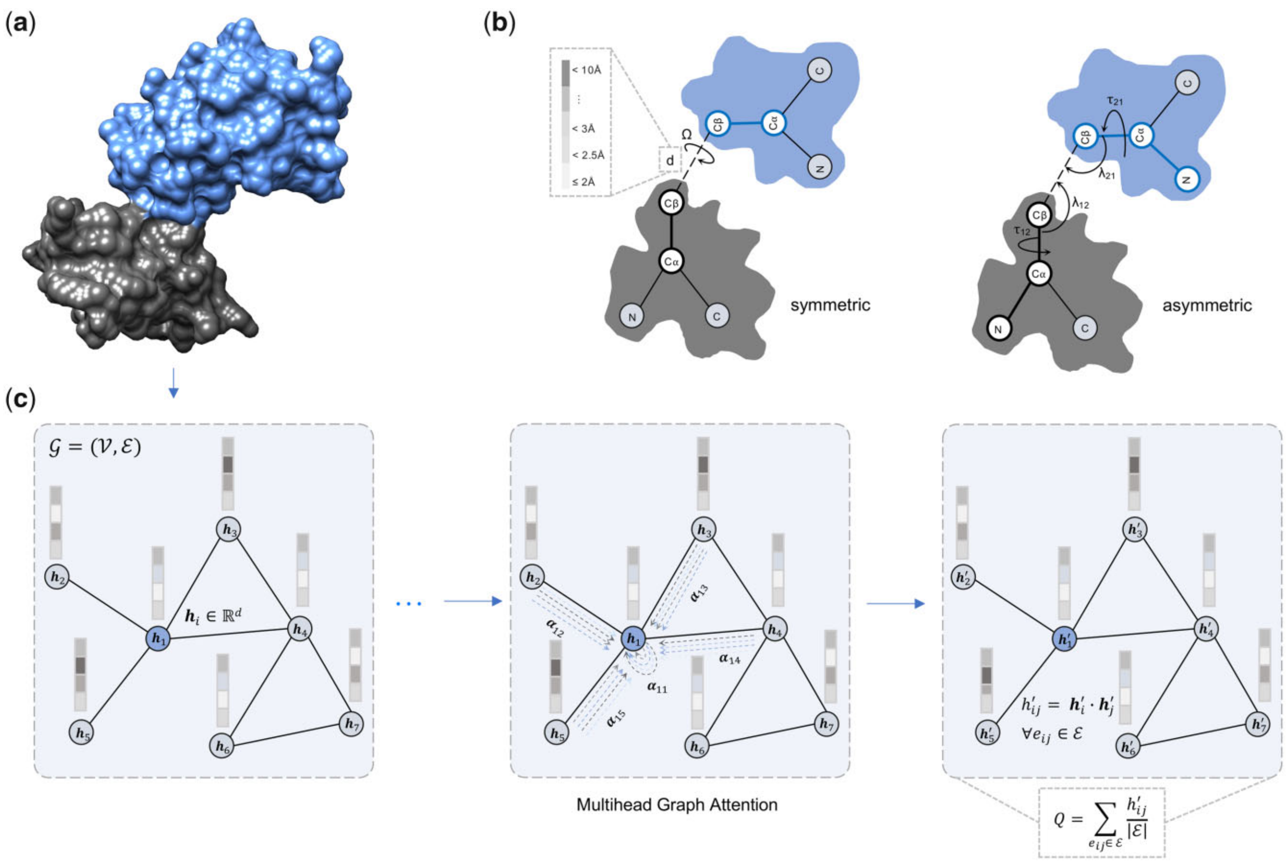
(a)
(b)
< 10Å
⋮
< 3Å
< 2.5Å
≤ 2Å
d
Ω
Cβ
Cα
C
N
symmetric
τ21
λ21
λ12
τ12
asymmetric
(c)
$\mathcal{G} = (\mathcal{V}, \mathcal{E})$
$\boldsymbol{h}_i \in \mathbb{R}^d$
h_1 h_2 h_3 h_4 h_5 h_6 h_7
...
α_{12} α_{13} α_{14} α_{11} α_{15}
Multihead Graph Attention
h'_1 h'_2 h'_3 h'_4 h'_5 h'_6 h'_7
$h'_{ij} = \boldsymbol{h}'_i \cdot \boldsymbol{h}'_j$
$\forall e_{ij} \in \mathcal{E}$
$Q = \sum_{e_{ij} \in \mathcal{E}} \frac{h'_{ij}}{|\mathcal{E}|}$

Figure 1. Illustration of the PIQLE framework for protein–protein interface quality estimation. (**a**) The predicted protein complex structure with its two interacting monomers colored in grey and blue. (**b**) Multimeric interaction geometries characterized by the inter-atomic distance and orientations of the residues at the interaction interface. (**c**) Graph representation of the interaction interface and quality estimation of individual interacting residues by edge-level error regression using multi-head GAT followed by a probabilistic combination

threshold of 10^{-3} for searching against the Uniclust30 (Mirdita *et al.*, 2017) database with a query sequence coverage of 10% and maximum pairwise sequence identity of 90% (Shuvo *et al.*, 2020). We then calculate the normalized number of sequences as:

$$N_{\mathrm{eff}}{}_{\mathrm{eff}}^{\mathrm{norm}} = \frac{N_{\mathrm{eff}}}{\surd \mathrm{L}},$$

where N_{eff} is the reciprocated sum of the number of sequences in the MSA having a sequence identity >80% to the *n*th sequence and *L* is the length of the sequence (Li *et al.*, 2019). We calculate the normalized number of effective sequences N_{eff} for each partner in the complex, following the approach described in Zhang *et al.* (2020). Additionally, we generate coupled MSA considering both the interacting partner using GLINTER (Xie and Xu, 2022), following the method described in ComplexContact (Zeng *et al.*, 2018) and compute N_{eff} of the coupled MSA. Therefore, each of the nodes in a graph has two evolutionarily features including the N_{eff} computed from the MSA for each partner in the complex and the N_{eff} calculated from the coupled MSA. The two evolutionarily features are considered as node features in the interface graph.

2.1.2 Edge features

Multimeric interaction distance: To capture multimeric interaction geometry, we discretize the Euclidian distance between the C_β atoms (C_α for glycine) of the interacting interface residue pairs into 17 bins ranging from 2 to 10 Å having a bin width of 0.5 Å. The discretized interaction distance is represented by one-hot encoding, resulting in 17 edge features.

Multimeric interaction orientation: In addition to C_β–C_β distances, we also include the orientations of the interacting interface residue pairs by extending the work of trRosetta (Yang *et al.*, 2020) for multimers. In particular, our multimeric interaction orientation is represented by three torsion (Ω, τ_{12} and τ_{21}) and two planar angles (λ_{12}, λ_{21}), as shown in Figure 1b. The Ω torsion angle measures the rotation along the virtual axis connecting the C_β atoms of the interacting interface residue pairs, and τ_{12}, λ_{12} (τ_{21}, λ_{21}) angles specify the direction of the C_β atom of interface residue of the first (second) interacting monomer in a reference frame centered on the interface residue of the second (first) interacting monomer. Unlike the symmetric torsion angle Ω, τ and λ are asymmetric and depend on the order of the monomeric interacting interface residue pairs. Once again, we perform sinusoidal and cosine transformations of the angles, leading to 10 features.

2.2 Network architecture

Figure 1c shows the architecture of our multi-head GAT for protein–protein interface quality estimation. The network consists of four multi-head graph attention layers (Veličković *et al.*, 2018). All the intermediate layers have four attention heads except for the output layer, which has one attention head. We perform hyperparameter selection on an independent validation set using grid search to determine the optimal number of layers and heads (Supplementary Table S2). The input layer of the network takes the interfacial graph G consisting of nodes V and edges E with the associated nodes and edge features as $G(V_i \in \mathbb{R}^{v\times 1} \times E_{ij} \ \in \mathbb{R}^{e\times 1})$. We use an empirically selected hidden dimension of 32 for the input layer with a scaling factor of 0.5 for each succeeding layer. Additionally, we perform a concatenation operation of all the heads along the output dimension of 1. Therefore, the output dimension of each intermediate layer depends on the number of hidden dimensions, and thus the output dimension of the multi-head attention layer l is as follows:

$$X^l = ||_{k=1}^4 h^l,$$

where k represents the number of heads and h is the hidden dimension at layer l. Of note, each multi-head attention layer performs a series of operations before it feeds the output to the next layer. First, we concatenate both the node and edge features and perform a linear transformation to embed both the node (h_i^l) and edge (e_{ij}^l) input features with the initialized weight W to d-dimensional hidden features assigned to each node (Dwivedi *et al.*, 2022; Veličković *et al.*, 2018) as follows:

$$z^l = W^l h_i^l e_{ij}^l,$$

where z represents the embedded features at layer l and W is the learnable network parameters, normalized using the Xavier weight initialization procedure at each layer to prevent vanishing and exploding gradient problems (Glorot and Bengio, 2010). We then compute an attention score a_{ij} between the neighboring nodes of each edge by performing self-attention on the incident nodes as:

$$a_{ij}^l = \sigma\Big(W(z_i^l|(z_j^l),$$

where z_i^l and z_j^l are the embeddings of the incident nodes of an edge. Both embeddings are concatenated, and a dot product is computed with a learnable weight vector W $\left(W \in \mathbb{R}^{\mathrm{D}}\right)$, where D represents the input dimension. Meanwhile, the node features of each node are updated with the combination of neighboring node features and the attention score a_{ij} as follows:

$$h_i^l = \ \sigma\left(\sum_{j\in N(i)} a_{ij}^l z_j^l\right).$$

2.3 Model training

For the assignment of the ground truth interface quality score during training, we first calculate the observed C_β–C_β distance between the interacting interface residue pairs in the predicted complex structural model (d_{ij}^{model}) and the corresponding residue pairs in the native structure (d_{ij}^{native}). We then assign a normalized ground truth interface quality score z_{ij} to the edge e_{ij} as follows:

$$z_{ij} = \begin{cases} 1 & if d_{ij}^{\mathrm{model}} \ < \ 10\text{Å and } d_{ij}^{\mathrm{native}} \ < \ 10\ \text{Å} \\ \dfrac{1}{1+\left(\frac{\left|d_{ij}^{\mathrm{model}}-d_{ij}^{\mathrm{native}}\right|}{d_0}\right)^2} & \text{otherwise,} \end{cases}$$

where $\left|d_{ij}^{\mathrm{model}} - d_{ij}^{\mathrm{native}}\right|$ is the observed edge-level error between the interacting interface residue pairs corresponding to the edge e_{ij}, and d_0 is a normalizing constant whose value is set to 10 Å to be consistent with the 10 Å threshold used for defining interacting residue pairs in the literature (Marze *et al.*, 2018).

During model training, we learn the interface quality score z_{ij} for each edge e_{ij} through edge-level error regression by optimizing the mean squared error loss function with sum reduction using the Deep Graph Library (Wang *et al.*, 2020a). We use the Adam optimizer (Kingma and Ba, 2014) with a learning rate of 0.001 and a weight decay of 0.0005. The training process consists of at most 500 epochs on an NVIDIA A40 GPU having an early stopping criterion with patience set to 40 to prevent overfitting.

2.4 Estimation of protein–protein interface quality

During the inference, we first estimate the interface quality score for each of the interacting residue pairs in the interface graph through edge-level error regression by computing the dot product between the predicted embeddings of the corresponding nodes as follows:

$$h'_{ij} = h'_i \ \times \ h'_j,$$

where h'_i and h'_j represent the node embeddings of nodes i and j connected by the edge e_{ij} in the final layer of the multi-head GAT. We then probabilistically combine the estimated quality scores of the individual interfacial residue pairs for estimating the overall interface quality score Q as follows:

$$Q = \sum_{e_{ij}\in\varepsilon} \frac{h'_{ij}}{|\varepsilon|},$$

where $|\varepsilon|$ represents the number of interfacial residue pairs in the model and h'_{ij} is the estimated interaction score for the edge e_{ij}. The overall interface quality score Q ranges between 0 and 1 with a higher score indicating better protein–protein interface quality.

2.5 Datasets

To train the GAT of PIQLE, we use the docking benchmark set of Dockground (Kundrotas *et al.*, 2018) version 2 (hereafter called Dockground v2) containing 179 dimeric protein complex targets having the length ranging from 92 to 894 residues with 100 complex structural models for each target, generated by docking the unbound structure of the receptor to the ligand.

To benchmark our method, we use the docking benchmark set of Dockground version 1 (hereafter called Dockground v1), comprising 61 dimeric protein complex targets having lengths ranging from 107 and 892 residues. We discard all targets from the Dockground v1 test dataset overlapping with our training set Dockground v2 using an average pairwise sequence identity cutoff of 20%, resulting in 23 dimer targets with an average of 109 decoys per target. Additionally, we use the Heterodimer-AF2 (hereafter called HAF2) (Chen *et al.*, 2023) dataset consisting of 13 targets having an average of 105 decoys per target with the length ranging from 78 to 1248 residues generated using AlphaFold-Multimer (Evans *et al.*, 2022).

For ablation studies, we use the docking Benchmark version 4.0 (Hwang *et al.*, 2010) comprising 69 dimeric protein complex targets having the length ranging from 23 to 822 residues that are non-overlapping to the targets in the training and benchmarking datasets (pairwise sequence identity cutoff of 20%) with each target having 100 complex structural models generated by ZDOCK (Pierce *et al.*, 2014). It is worth noting that all the datasets used for training, benchmarking and ablation studies are non-overlapping with an average pairwise sequence identity of <20% between any pair of datasets (Supplementary Table S3).

2.6 Evaluation metrics and competing methods

We assess the performance of our method using various evaluation metrics based on the DockQ scores (Basu and Wallner, 2016). DockQ score integrates various CAPRI measures including F_{Nat}, LRMS and iRMS to evaluate the quality of protein–protein docking models (Lensink and Wodak, 2013). F_{Nat} is defined by the fraction of native interfacial contacts in the model. The root mean square deviations (RMS) for the ligand (LRMS) and interface (iRMS) between the model and the target are calculated as follows:

$$\text{RMS}_{\text{scaled}}(\text{RMS}, d_i) = \frac{1}{1 + \left(\frac{\text{RMS}}{d_i}\right)^2},$$

where d_i represents the scaling factors: d_1 for LRSM and d_2 for iRMS. d_1 and d_2 are optimized to be 8.5 and 1.5 Å, respectively, based on the ability to separate models according to CAPRI classifications in terms of $F1$ scores (Basu and Wallner, 2016; Lensink and Wodak, 2013). Finally, the DockQ score is calculated by combining the aforementioned scoring terms as follows:

$$\text{DockQ}(F_{\text{nat}},\ \text{LRMS},\ \text{iRMS},\ d_1, d_2) = \frac{(F_{\text{nat}} +\ \text{RMS}_{\text{scaled}}(\text{LRMS}, d_1) +\ \text{RMS}_{\text{scaled}}(\text{iRMS}, d_2))}{3}.$$

DockQ scores range between 0 and 1 with a higher score indicating better model quality. Using the DockQ scores as the ground truth, we employ three evaluation criteria to measure the protein–protein interface quality estimation performance: (i) ability to reproduce the ground truth DockQ scores, (ii) ability to rank complex structural models and (iii) ability to distinguish acceptable from incorrect models. For the first criterion, we use the Spearman correlation coefficient (ρ) between the estimated quality of the protein complexes and their corresponding DockQ scores. Consequently, a higher correlation indicates better reproducibility. For the second criterion, we use the top-N success rate (herein: SR) and top-N hit rate (herein: HR) (Guo *et al.*, 2022). The top-N success rate is calculated as the percentage of complex targets having at least one acceptable model among top-N ranked models as follows:

$$\text{SR}(N) = \frac{S(N)}{K} \times 100\%,$$

where $S(N)$ is the number of complex targets having at least one acceptable model among top-N ranked models and K is the total number of targets, where the standard cutoff of DockQ = 0.23 is used to identify acceptable models (Bryant *et al.*, 2022). The top-N hit rate is calculated as the fraction of acceptable models among top-ranked models relative to all acceptable models in the entire dataset as follows:

$$\text{HR}(N) = \frac{H(N)}{M} \times 100\%,$$

where $H(N)$ is the total number of acceptable models among top-N ranked models and M is the total number of acceptable models in the dataset. Higher success and hit rate indicate better ranking ability, especially for low values of N. We evaluate the success and hit rate of top-ranked models for various values of N including top-1, top-5, top-10, top-15, top-20, top-25 and top-30. We further evaluate the methods' ranking performance on the top-N ranked models after categorizing them into acceptable-, medium- and high-quality complex models based on their DockQ scores using standard CAPRI criteria. For the third criterion, we perform receiver operating characteristics (ROC) analysis using a DockQ score cutoff of 0.23 to separate acceptable and incorrect models. Meanwhile, the area under the ROC curve (AUC) quantifies the ability of a method to distinguish between acceptable and incorrect models with a higher AUC indicating better distinguishing ability.

We compare the performance of PIQLE against a number of existing protein complex quality estimation methods ranging from physics-based approaches to machine learning-based methods. As a representative physics-based method, we compare PIQLE against ZRANK2 (Pierce and Weng, 2008), which is an improved version of ZRANK (Pierce and Weng, 2007). For a fair comparison, we use a min.–max. normalization strategy to scale ZRANK2 energy scores to the same range as the predicted scores of other methods including PIQLE as follows:

$$\text{ZRANK2} = \frac{X - X_{\max}}{X_{\min} - X_{\max}},$$

where X is the raw ZRANK2 estimated energy scores and $X_{\min}$ and $X_{\max}$ represent the smallest and largest estimated scores, respectively, considering all predicted complex structural models for a specific target.

We also compare the performance of PIQLE against various machine learning-based approaches including 3DCNN-based methods TRScore (Guo *et al.*, 2022) and four variants of DOVE (Wang *et al.*, 2020b): DOVE-Atom20, DOVE-Atom40, DOVE-GOAP and DOVE-Atom40+GOAP as well as recent GNN-based methods GNN-DOVE (Wang *et al.*, 2021) and DProQA (Chen *et al.*, 2023). We exclude the comparison to GNN-DOVE and TRScore on the Dockgorund v1 test dataset due to the overlap of the training datasets used in GNN-DOVE and TRScore with the complex targets present in the Dockground v1 dataset. Meanwhile, all methods are included for performance comparison on the HAF2 test dataset.

3 Results

3.1 Reproducing ground truth DockQ scores

Figure 2 shows the Spearman correlation coefficients (ρ) between the estimated qualities of the protein–protein interfaces and their corresponding ground truth DockQ scores for PIQLE and the other competing methods. PIQLE consistently outperforms all other competing methods in both Dockground v1 and HAF2 datasets. On Dockground v1, PIQLE attains the highest Spearman correlation of 0.519,

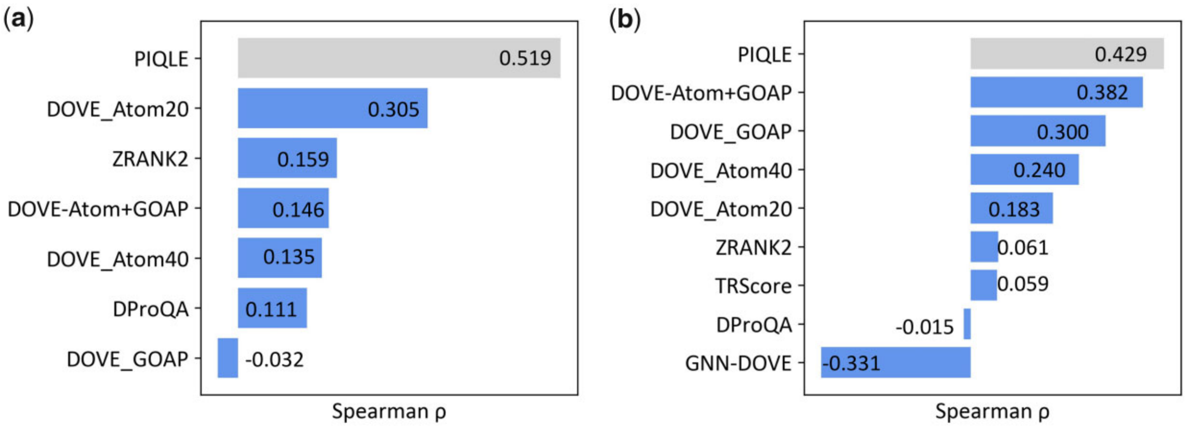

Figure 2. Reproducibility of ground truth DockQ scores for PIQLE (in gray) and the competing methods (blue), sorted in decreasing order of Spearman correlations coefficient (ρ) between the estimated qualities of the protein–protein interfaces and their corresponding DockQ scores on (**a**) Dockground v1 and (**b**) HAF2 datasets

which is much better than the second-best 3DCNN-based method DOVE-Atom20 (0.305), the recent graph transformer network (GTN)-based method DProQA (0.111) and physics-based scoring function ZRANK2 (0.159). The same trend continues for the HAF2 dataset, in which PIQLE attains the highest Spearman correlation of 0.429, which is significantly better than the other competing methods. The 3DCNN-based method DOVE remains the second-best method with its variant DOVE-Atom+GOAP attaining a Spearman correlation of 0.382. The recent GNN-based methods DProQA and GNN-DOVE, however, fail to generalize on the HAF2 dataset attaining negative Spearman correlations of −0.015 and −0.331, respectively. In summary, our method PIQLE exhibits an improved ability to reproduce ground truth DockQ scores with high fidelity.

3.2 Ranking complex structural models

Figure 3a and b shows the complex model ranking performance of PIQLE and the other competing methods in terms of the success rate (SR) metric, which evaluates the ability of a method to select at least one acceptable model within top-*N* ranked models. As shown in Figure 3a, PIQLE consistently achieves the highest SR among all methods for almost all top-*N* rankings in the Dockground v1 dataset. The noticeably higher SR of PIQLE at low values of *N*, such as top-1 (∼44%) and top-5 (∼74%), is particularly noteworthy. In the HAF2 dataset (Fig. 3b), PIQLE attains a higher top-1 SR of ∼77% compared to the other methods, whereas some of the other methods, such as ZRANK2 and DOVE-ATOM20, achieve comparable or higher SR values, particularly for high values of *N*. Overall, PIQLE frequently attains higher success rates, particularly when *N* is low.

Figure 3c and d shows the ranking ability of PIQLE and the other competing methods in terms of the hit rate (HR) metric, which evaluates the performance of a method based on the total number of acceptable models among top-ranked models relative to all acceptable models in the entire dataset. As shown in Figure 3c, PIQLE significantly outperforms all other competing methods by achieving the highest HR on Dockground v1 dataset, for all values of *N*. For example, PIQLE improves the top-10 HR by more than 30% over the second-best method DProQA (37.079 versus 28.125). PIQLE also consistently attains better HR performance on the HAF2 dataset as shown in Figure 3d. It is interesting to note the somewhat low HR performance of all methods including ours on the HAF2 dataset. While all methods, particularly the machine learning-based approaches achieve higher SR on the HAF2 dataset, they appear to be less effective at selecting a large proportion of acceptable models from a smaller number of top-ranked models measured by HR, suggesting a need for further improvement. Nevertheless, our new method PIQLE strikes an ideal balance to deliver top performance in terms of both success rate and hit rate across different datasets, indicating its all-round ability in ranking complex structural models.

We further benchmark the ranking performance of PIQLE, and other competing methods on the same test datasets based on the standard CAPRI criteria of acceptable-, medium- and high-quality complex models in terms of DockQ scores (Supplementary Figs S1–S3). Overall, PIQLE delivers a well-rounded ranking performance considering both success and hit rates metrics across various values of *N* for acceptable-, medium- and high-quality complex models.

3.3 Distinguishing acceptable from incorrect models

In addition to reproducing the ground truth DockQ scores with high fidelity and accurately ranking complex structural models, the ability to distinguish acceptable from non-acceptable prediction is critically important. Figure 4 shows the AUC attained by PIQLE and the competing methods on the test datasets. PIQLE consistently outperforms all other competing methods by achieving the best AUC on both the Dockground v1 and HAF2 datasets. For the Dockground v1 dataset, AUC attained by PIQLE is 0.711, which is closely followed by the second-best performing method DOVE-Atom40+GOAP with an AUC of 0.710. On the HAF2 dataset, PIQLE attains an AUC of 0.743, which is significantly higher than all competing methods including the second-best method DOVE-Atom40+GOAP having an AUC of 0.648. In summary, PIQLE exhibits an improved ability to distinguish between acceptable and non-acceptable prediction.

3.4 Case study

Figure 5 shows some representative examples of protein–protein interface quality estimation by PIQLE for selected targets from Dockground v1 and HAF2 datasets with varying degrees of predictive modeling accuracy. For a reasonably well predicted complex structural model for Dockground v1

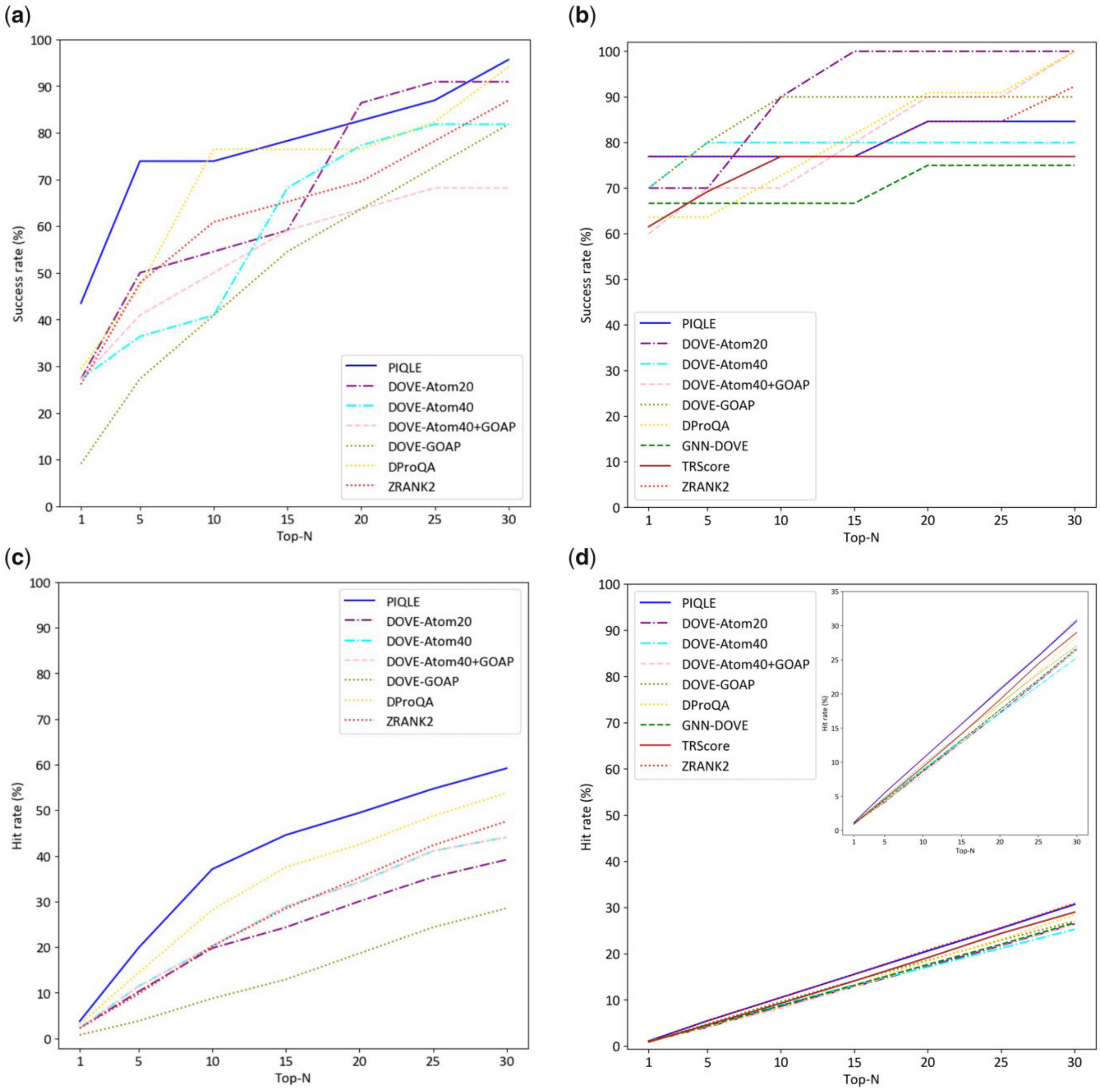

Figure 3. Ranking complex structural models for PIQLE and the competing methods in terms of success rate on (**a**) Dockground v1 dataset, (**b**) HAF2 dataset and hit rate on (**c**) Dockground v1 dataset, (**d**) HAF2 dataset based on top-1, top-5, top-10, top-15, top-20, top-25 and top-30 models. A cutoff of DockQ =0.23 is used to identify acceptable models

target 1r0r having a DockQ score of 0.732 (Fig. 5a), PIQLE estimates an interfacial quality score of 0.625. Apart from a few false positive interacting residue pairs, most of the interface regions in this predicted complex structural model are correct with an $F1$ score of 0.674 considering the previously defined C_β–C_β distance threshold of 10 Å (Marze *et al.*, 2018) for identifying the true interacting residue pairs. For a moderate quality predicted complex structural model for HAF2 target 7nkz having a DockQ score of 0.478 and several false positive interacting residue pairs with an $F1$ score of 0.333 (Fig. 5b), PIQLE estimates a moderate interfacial quality score of 0.398. Additionally, Figure 5c and d shows two low-quality predicted complex structural models for Dockground v1 target 1ppf and HAF2 target 7lxt having a DockQ score of 0.102 and 0.127, respectively, with noticeably wrong interfaces. For these models, PIQLE estimates much lower interfacial quality scores of 0.143 and 0.130, respectively.

3.5 Ablation study

To examine the relative importance of the features adopted in PIQLE, we conduct feature ablation experiments by gradually isolating the contribution of individual feature or groups of features during model training and evaluating the accuracy on the independent ZDOCK validation dataset. Figure 6a shows the Spearman correlation coefficients (ρ) between the estimated qualities of the protein–protein interfaces and their corresponding ground truth DockQ scores when various features are isolated from the full-fledged version of PIQLE. The results demonstrate that all features contribute to the overall performance achieved by PIQLE. For example, we notice an

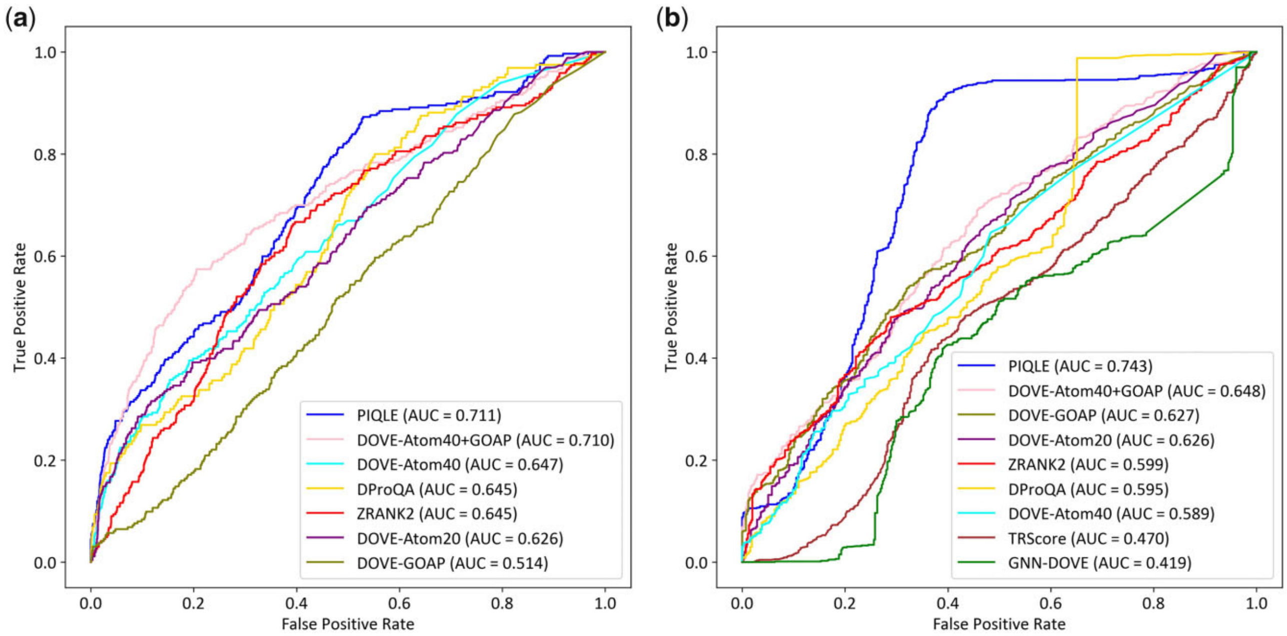

Figure 4. Distinguishability of acceptable vs. incorrect models for PIQLE and the competing methods on (**a**) Dockground v1 and (**b**) HAF2 datasets. A cutoff of DockQ =0.23 is used to separate acceptable from incorrect models

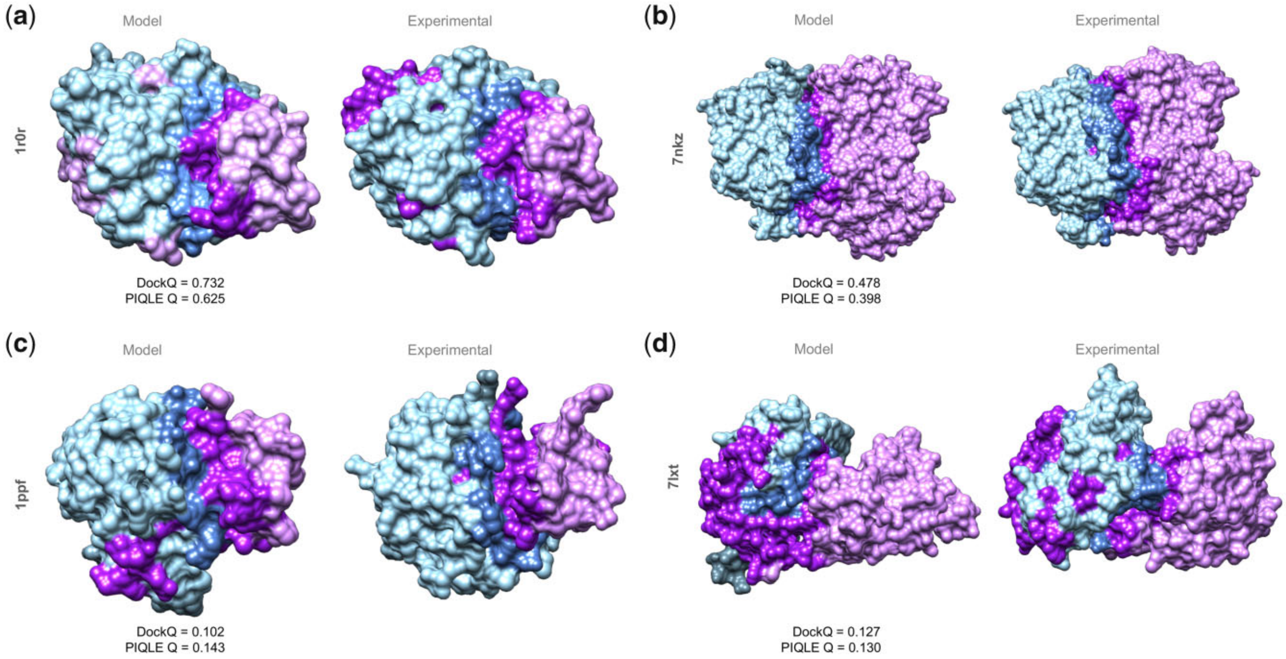

Figure 5. Case study on protein–protein interface quality estimation by PIQLE using predicted complex structural models for (**a**) Dockground v1 target 1r0r, (**b**) HAF2 target 7nkz, (**c**) Dockground v1 target 1ppf and (**d**) HAF2 target 7lxt. For each target, the interacting protein chains are colored in blue (chain 1) and purple (chain 2) with the interface regions highlighted in darker shades of blue and purple. The experimental structures for each target are shown side-by-side with the observed interface regions annotated

accuracy decline when we isolate the sequence-based features one by one including amino acid residue encoding (no residue encoding) and relative residue positioning (no relative residue positioning). Importantly, discarding evolutionarily information noticeably declines the overall performance (no evolutionarily information), indicating the effectiveness of MSA-derived evolutionarily information. Not surprisingly, we notice a dramatic performance drop when both the residue-based features and the evolutionary information are isolated (no residue+evolutionarily information). Similarly, we also notice a performance drop when the feature based on local backbone geometry is discarded (no local backbone geometry). Additionally, we notice a consistent accuracy decline when we discard the newly introduced edge features based on multimeric interaction distance (no multimeric interaction distance), multimeric orientation (no multimeric interaction orientation) and their combination (no multimeric interaction geometry). That is, the improved performance of our method is connected to the effective integration of multimeric interaction geometries.

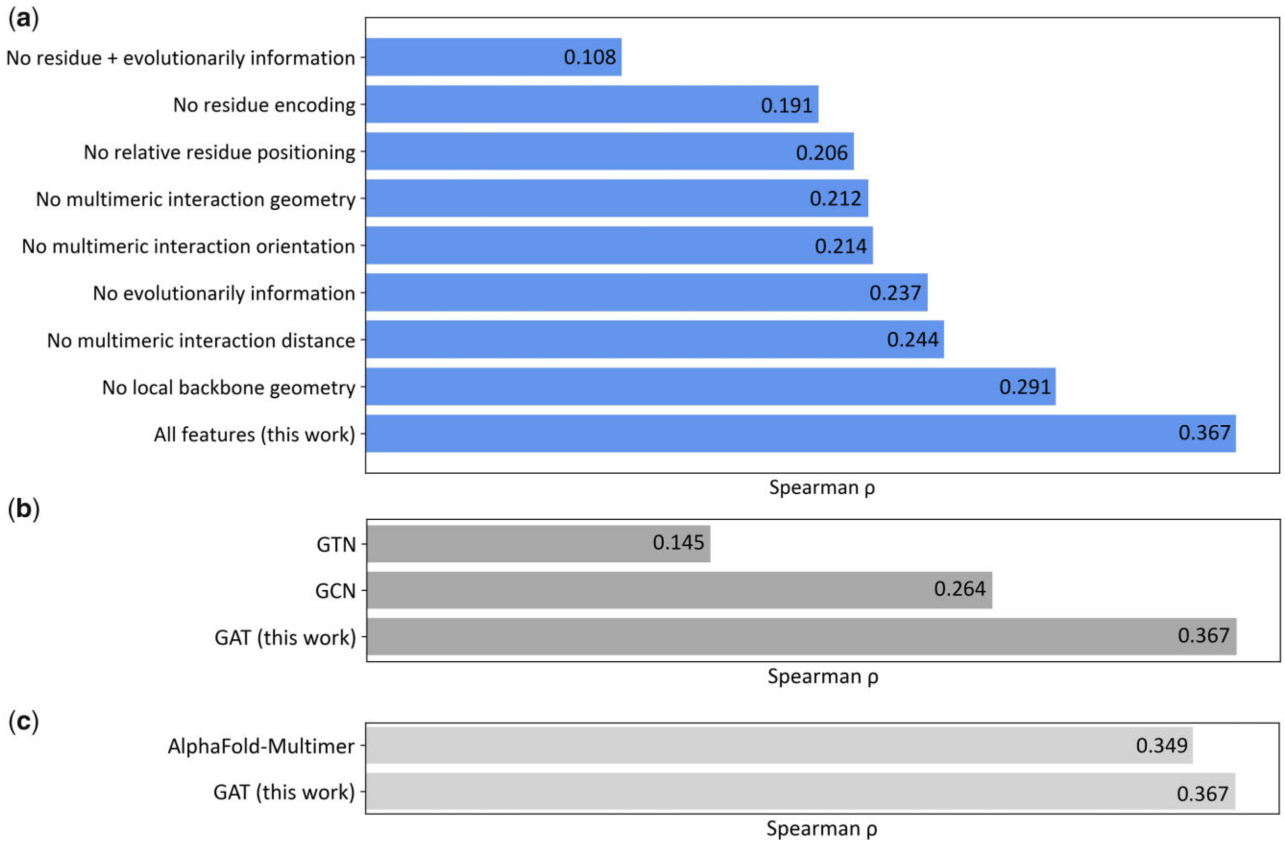

Figure 6. Ablation study on the independent ZDOCK validation dataset in terms of Spearman correlations coefficient (ρ) between the estimated qualities of the protein–protein interfaces and their corresponding DockQ scores by (**a**) gradually isolating individual feature or groups of features during model training, (**b**) training two baseline GNN models employing GCN and GTN architectures and (**c**) the performance comparison between GAT employed in PIQLE and the ipTM scores predicted by the self-assessment module of AlphaFold-Multimer repurposed for protein complex scoring

To further investigate the contribution of the multi-head GAT model used in PIQLE, we train two baseline GNN-based models for protein–protein interface quality estimation: graph convolutional network (GCN) (Kipf and Welling, 2017) and GTN (Yun *et al.*, 2019). All baseline networks are trained on the same training dataset using the same set of input features as the full-fledged version of PIQLE. Following the same approach as used for PIQLE's GAT, we perform hyperparameter selection for GCN and GTN on the same independent validation set by varying the number of layers and heads for GTN and the number of layers for GCN (Supplementary Table S2). Figure 6b shows the performance of PIQLE compared to the baseline networks on the independent ZDOCK validation dataset in terms of the Spearman correlation coefficients (ρ) between the estimated qualities of the protein–protein interfaces and their corresponding ground truth DockQ scores. The multi-head GAT architecture of PIQLE significantly outperforms the other baseline networks, demonstrating its effectiveness for protein–protein interface quality estimation task.

We further compare the performance of PIQLE with the interface predicted TM scores (ipTM) predicted by the self-assessment module of AlphaFold-Multimer (Evans *et al.*, 2022) repurposed for protein complex scoring (Roney and Ovchinnikov, 2022). It is important to note that ipTM is a self-assessment score generated by AlphaFold-Multimer for estimating the accuracy of their own predicted complex structural models in terms of the quality of the multimeric interaction interface. As such, the ipTM scores predicted by AlphaFold-Multimer are not equivalent to the interface quality scores estimated from an independent protein complex scoring method, such as PIQLE. Nonetheless, the comparison may offer some interesting insights. We utilize an extended version of the AF2Rank method (Roney and Ovchinnikov, 2022) repurposed for protein complex scoring based on the self-assessment module of AlphaFold-Multimer, freely available as a Google Colab Notebook at https://colab.research.google.com/github/sokrypton/ColabDesign/blob/main/af/examples/AF2Rank.ipynb as of February 26, 2023, for generating the ipTM scores. Figure 6c shows the performance in terms of Spearman correlation coefficient (ρ) on the ZDOCK set for the estimated interface quality scores by PIQLE and the ipTM scores predicted by the self-assessment module of AlphaFold-Multimer repurposed for protein complex scoring. PIQLE convincingly outperforms (ρ = 0.367) the repurposed self-assessment complex scoring of AlphaFold-Multimer (ρ = 0.349), even though the feature ablated variants of PIQLE (Fig. 6a) as well as the baseline GNNs GTN and GCN (Fig. 6b) fall short. The results further demonstrate the contribution of both the network architecture and features used in PIQLE for improved protein–protein interface quality estimation performance beyond what is attainable by the self-assessment module of AlphaFold-Multimer repurposed for protein complex scoring.

4 Conclusion

This work introduces PIQLE, a new method for protein–protein interface quality estimation by deep graph learning of multimeric interaction geometries. PIQLE exploits multi-head GAT architecture leveraging multimeric interaction geometries and evolutionarily information along with sequence- and structure-derived features to estimate the quality of the individual interactions between the interfacial residues and then probabilistically combines the estimated quality of the interfacial residues for scoring the overall interface. We demonstrate that PIQLE attains state-of-the-art protein–protein interface quality estimation performance by conducting large-scale benchmarking on multiple widely used protein docking decoy sets. Our ablation study and comparison with the self-assessment module of AlphaFold-Multimer repurposed for protein complex scoring on an independent validation set confirm the contribution of various features adopted in PIQLE and the effectiveness of the multi-head GAT architecture.

Our study leads to a number of future directions to consider: of particular interest is the possibility of broadening the applicability of our method for higher order oligomers and large protein assemblies. Further, a promising direction for future work is to consider the diversity of predictive modeling ensemble and conformational states of the interacting monomers for interface quality estimation for interacting proteins having multi-state conformational dynamics. Finally, integrating complementary features, such as residue-level self-assessment confidence estimates for the interacting protein chains and sequence-based disorder prediction coupled with a richer deep graph representation learning framework may further boost protein–protein interface quality estimation performance. We expect our method to be extended to other biomolecular interface characterization, including estimating the quality of predicted protein interaction with other molecules, such as DNA, RNA and small ligands.

Funding

This work was partially supported by the National Institute of General Medical Sciences [R35GM138146 to D.B.]; and the National Science Foundation [DBI2208679 to D.B.].

Conflict of Interest

none declared.

Data availability

The raw data used in this study, including the datasets for train, test and validation are collected from publicly available sources. The Dockground v2 training set is available at https://dockground.compbio.ku.edu/downloads/unbound/decoy/decoys-set2-1.0.tgz. The Dockground v1 test set is available at https://dockground.compbio.ku.edu/downloads/unbound/decoy/decoys1.0.zip. The Heterodimer-AlphaFold2 test set is available at https://zenodo.org/record/6569837/files/DproQ_benchmark.tgz. The ZDOCK docking benchmark version 4.0 validation set is available at http://zlab.umassmed.edu/benchmark/.

References

Andreani,J. *et al.* (2013) InterEvScore: a novel coarse-grained interface scoring function using a multi-body statistical potential coupled to evolution. *Bioinformatics*, **29**, 1742–1749.

Baek,M. *et al.* (2021) Accurate prediction of protein structures and interactions using a three-track neural network. *Science*, **373**, 871–876.

Basu,S. and Wallner,B. (2016) DockQ: a quality measure for protein-protein docking models. *PLoS One*, **11**, e0161879.

Bryant,P. *et al.* (2022) Improved prediction of protein-protein interactions using AlphaFold2. *Nat. Commun.*, **13**, 1265.

Cao,Y. and Shen,Y. (2020) Energy-based graph convolutional networks for scoring protein docking models. *Proteins*, **88**, 1091–1099.

Chen,X. *et al.* (2023) A gated graph transformer for protein complex structure quality assessment and its performance in CASP15. bioRxiv 2022.05.19.492741.

Christoffer,C. *et al.* (2021) LZerD protein-protein docking webserver enhanced with de novo structure prediction. *Front. Mol. Biosci.*, **8**, 724947.

Dai,B. and Bailey-Kellogg,C. (2021) Protein interaction interface region prediction by geometric deep learning. *Bioinformatics*, **37**, 2580–2588.

Dwivedi,V.P. *et al.* (2022) Benchmarking graph neural networks. *arXiv preprint* arXiv: 2003.00982v5.

Evans,R. *et al.* (2022) Protein complex prediction with AlphaFold-Multimer. bioRxiv 2021.10.04.463034.

Ganea,O.-E. *et al.* (2022) Independent SE(3)-equivariant models for end-to-end rigid protein docking. *arXiv preprint* arXiv: 2111.07786v2.

Glorot,X. and Bengio,Y. (2010) Understanding the difficulty of training deep feedforward neural networks. In: *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics. JMLR Workshop and Conference Proceedings*, Sardinia, Italy, pp. 249–256.

Guo,L. *et al.* (2022) TRScore: a 3D RepVGG-based scoring method for ranking protein docking models. *Bioinformatics*, **38**, 2444–2451.

Huang,S.-Y. and Zou,X. (2008) An iterative knowledge-based scoring function for protein-protein recognition. *Proteins*, **72**, 557–579.

Hwang,H. *et al.* (2010) Protein–protein docking benchmark version 4.0. *Proteins*, **78**, 3111–3114.

Jumper,J. *et al.* (2021) Highly accurate protein structure prediction with AlphaFold. *Nature*, **596**, 583–589.

Kabsch,W. and Sander,C. (1983) Dictionary of protein secondary structure: pattern recognition of hydrogen-bonded and geometrical features. *Biopolymers*, **22**, 2577–2637.

Kingma,D.P. and Ba,J. (2014) Adam: a method for stochastic optimization. *arXiv preprint* arXiv: 1412.6980v9.

Kipf,T.N. and Welling,M. (2017) Semi-supervised classification with graph convolutional networks. *arXiv preprint* arXiv: 1609.02907v4.

Kumar,K. *et al.* (2018) Cation–π interactions in protein–ligand binding: theory and data-mining reveal different roles for lysine and arginine. *Chem. Sci.*, **9**, 2655–2665.

Kundrotas,P.J. *et al.* (2018) Dockground: a comprehensive data resource for modeling of protein complexes. *Protein Sci.*, **27**, 172–181.

Lensink,M.F. and Wodak,S.J. (2013) Docking, scoring, and affinity prediction in CAPRI. *Proteins*, **81**, 2082–2095.

Li,H. *et al.* (2017) Deep learning methods for protein torsion angle prediction. *BMC Bioinformatics*, **18**, 417.

Li,Y. *et al.* (2019) ResPRE: high-accuracy protein contact prediction by coupling precision matrix with deep residual neural networks. *Bioinformatics*, **35**, 4647–4655.

Lyskov,S. and Gray,J.J. (2008) The RosettaDock server for local protein-protein docking. *Nucleic Acids Res.*, **36**, W233–W238.

Marze,N.A. *et al.* (2018) Efficient flexible backbone protein–protein docking for challenging targets. *Bioinformatics*, **34**, 3461–3469.

Mirdita,M. *et al.* (2017) Uniclust databases of clustered and deeply annotated protein sequences and alignments. *Nucleic Acids Res.*, **45**, D170–D176.

Peng,X. *et al.* (2017) Protein-protein interactions: detection, reliability assessment and applications. *Brief. Bioinform.*, **18**, 798–819.

Pierce,B. and Weng,Z. (2007) ZRANK: reranking protein docking predictions with an optimized energy function. *Proteins*, **67**, 1078–1086.

Pierce,B. and Weng,Z. (2008) A combination of rescoring and refinement significantly improves protein docking performance. *Proteins*, **72**, 270–279.

Pierce,B.G. *et al.* (2014) ZDOCK server: interactive docking prediction of protein–protein complexes and symmetric multimers. *Bioinformatics*, **30**, 1771–1773.

Remmert,M. *et al.* (2011) HHblits: lightning-fast iterative protein sequence searching by HMM-HMM alignment. *Nat. Methods*, **9**, 173–175.

Roney,J.P. and Ovchinnikov,S. (2022) State-of-the-art estimation of protein model accuracy using AlphaFold. *Phys. Rev. Lett.*, **129**, 238101.

Shuvo,M.H. *et al.* (2020) QDeep: distance-based protein model quality estimation by residue-level ensemble error classifications using stacked deep residual neural networks. *Bioinformatics*, **36**, i285–i291.

Tavafoghi,M. and Cerruti,M. (2016) The role of amino acids in hydroxyapatite mineralization. *J. R Soc. Interface*, **13**, 20160462.

Vajda,S. *et al.* (2013) Sampling and scoring: a marriage made in heaven. *Proteins*, **81**, 1874–1884.

Veličković,P. *et al.* (2018) Graph attention networks. *arXiv preprint* arXiv: 1710.10903.

Wallner,B. (2022) AFsample: improving multimer prediction with AlphaFold using aggressive sampling. bioRxiv 2022.12.20.521205.

Wang,M. *et al.* (2020a) Deep Graph Library: a graph-centric, highly-performant package for graph neural networks. *arXiv preprint* arXiv: 1909.01315v2.

Wang,X. *et al.* (2020b) Protein docking model evaluation by 3D deep convolutional neural networks. *Bioinformatics*, **36**, 2113–2118.

Wang,X. *et al.* (2021) Protein docking model evaluation by graph neural networks. *Front. Mol. Biosci.*, **8**, 647915.

Xie,Z. and Xu,J. (2022) Deep graph learning of inter-protein contacts. *Bioinformatics*, **38**, 947–953.

Yang,J. *et al.* (2020) Improved protein structure prediction using predicted interresidue orientations. *Proc. Natl. Acad. Sci. USA*, **117**, 1496–1503.

Yun,S. *et al.* (2019) Graph transformer networks. In: *Proceedings of the 33rd International Conference on Neural Information Processing Systems*. pp. 11983–11993. Curran Associates Inc., Red Hook, NY, USA.

Zahiri,J. *et al.* (2020) Protein complex prediction: a survey. *Genomics*, **112**, 174–183.

Zeng,H. *et al.* (2018) ComplexContact: a web server for inter-protein contact prediction using deep learning. *Nucleic Acids Res.*, **46**, W432–W437.

Zhang,C. *et al.* (2020) DeepMSA: constructing deep multiple sequence alignment to improve contact prediction and fold-recognition for distant-homology proteins. *Bioinformatics*, **36**, 2105–2112.

Zhou,H. and Skolnick,J. (2011) GOAP: a generalized orientation-dependent, all-atom statistical potential for protein structure prediction. *Biophys. J.*, **101**, 2043–2052.

Zhou,J. *et al.* (2020) Graph neural networks: a review of methods and applications. *AI Open*, **1**, 57–81.

Zhu,C. *et al.* (2016) Characterizing hydrophobicity of amino acid side chains in a protein environment via measuring contact angle of a water nanodroplet on planar peptide network. *Proc. Natl. Acad. Sci. USA*, **113**, 12946–12951.