EGAT: Edge Aggregated Graph Attention Networks and Transfer Learning Improve Protein-Protein Interaction Site Prediction

Sazan Mahbub[1] and Md Shamsuzzoha Bayzid[1,*]

[1]Department of Computer Science and Engineering
Bangladesh University of Engineering and Technology
Dhaka-1205, Bangladesh
[*]Corresponding author: shams_bayzid@cse.buet.ac.bd

Abstract

Motivation: Protein-protein interactions are central to most biological processes. However, reliable identification of protein-protein interaction (PPI) sites using conventional experimental methods is slow and expensive. Therefore, great efforts are being put into computational methods to identify PPI sites.

Results: We present EGAT, a highly accurate deep learning based method for PPI site prediction, where we have introduced a novel edge aggregated graph attention network to effectively leverage the structural information. We, for the first time, have used transfer learning in PPI site prediction. Our proposed edge aggregated network, together with transfer learning, has achieved remarkable improvement over the best alternate methods. Furthermore, EGAT offers a more interpretable framework than the typical black-box deep neural networks.

Availability: EGAT is freely available as an open source project at `https://github.com/Sazan-Mahbub/EGAT`.

Keywords: Protein-Protein Interaction Sites, Deep Learning, Graph Neural Network, Edge Aggregation.

1 Introduction

Proteins are responsible of various functions in cells, but carrying out many of these function requires the interaction of more than one protein molecules [1]. This makes protein-protein interaction (PPI) one of the key elements for understanding underlying biological processes, including functions of the cells [2, 3], PPI networks [1, 4], disease mechanisms [5, 6], as well as for designing and developing novel therapeutics [7, 8].

Protein-protein interaction (PPI) sites are the interfacial residues of a protein that interact with other protein molecules. Several wet-lab methods, including two-hybrid screening and affinity purification coupled to mass spectrometry are usually used to identify PPI sites [9, 10, 11]. However, the experimental determination of PPI sites is costly and time- and labour-intensive [12, 13, 14]. Thus, highly accurate computational prediction methods can be a useful guide for and complement to genetic and biochemical experiments. Therefore, in the last two decades, computational approaches have emerged as an important means of predicting PPI sites [13, 15, 16, 17]. These computational methods can be roughly divided into three categories [18]: (1) Protein-protein docking and modeling [19], (2) Structure-based methods [15, 16], and (3) Sequence based methods [20, 21]. Docking and structure based methods, unlike the sequence-based methods, leverage the structural information of the protein molecules.

There are two major areas in in protein-protein interaction sites prediction (PPISP). One is pair-wise interaction sites prediction for predicting interfacial residues of a pair of proteins, which is related to the docking of two proteins. The second prediction problem – and the one addressed in this study – is the prediction of putative interaction sites upon the surface of an isolated protein, known to be involved in protein-protein interactions, but where the structure of the partner or complex is not known [22]. The absence of any information about the partner proteins makes the latter problem relatively more difficult and challenging [23, 24].

In order to predict the interfacial residues of a single (isolated) protein, most of the recent computational methods have applied various machine learning (ML) algorithms [15, 16, 18, 20, 25, 26, 27, 28, 29]. Many studies have shown the importance of using local contextual features for predicting interfacing residues [15, 18, 30, 31, 32], which is usually encoded by a sliding window with a fixed size. However, similar to other residue-level prediction problems, such as secondary structure, backbone torsion angle, relative accessible surface area [33, 34], information about the *long-range interactions* between the residues that are not sequentially closer but within a close proximity in three-dimensional Euclidean space is also very crucial for PPI sites prediction. Zeng

et al. [15] addressed this issue with global features, extracted using two-dimensional convolutional neural networks. But such global feature extraction method generates the same global feature representation for all the residues of a protein, and thus lacks the ability to learn various suitable functions for different residues, and subsequently may not effectively encode long-range interactions between different residues.

Position-Specific Scoring Matrix (PSSM) is one of the most useful features for the prediction of PPI sites [15], which is also widely used by many state-of-the-art methods [15, 16, 20]. Other features, such as primary sequence and secondary structure information, alone do not yield good predictions when PSSM features are not used [15]. Generation of PSSM using PSI-BLAST [35], however, is a time-consuming process, which Zhang *et al.* [21] pointed out as a major bottleneck of these methods. Hence, identifying the feature-sets, which are less computationally demanding yet effective for highly accurate prediction of PPI sites is of great interest. The recent advancement in Natural Language Processing (NLP) can contribute in this direction as a trained language model can extract features to use as input for a subsequently trained supervised model through transfer-learning [36]. In a very recent study, Elnaggar *et al.* [36] developed ProtTrans, which provides an outstanding model for protein pretraining. They showed that a new set of feature, generated by pre-trained transformer-like models are capable of performing very well while taking significantly less time compared to PSSM. They trained two auto regression language models (Transformer-XL [37] and XLNet [38]) and two auto encoding models (BERT [39] and Albert [40]) on data containing up to 393 billion amino acids from 2.1 billion protein sequences in a self-supervised manner, considering each residue as a "word" (similar to language modeling in natural language processing [39]). In a similar study, Vig *et al.* [41] showed that "attention scores" in some of the attention matrices of such pretrained transformer-like models correlate in various degrees with protein-attributes, including contact maps and certain types of binding sites.

Most of the existing methods for predicting PPI use features derived from the primary sequences of the proteins [15, 16, 18, 20, 21, 25, 27, 28, 29]. However, using only primary sequence based features may limit the capabilities of the methods to achieve higher accuracy [27]. Thus, several methods have leveraged features derived from structural data (generally from the PDB files [42]), including three state and eight state secondary structures [15, 16], the level of surface exposure in unbound structures [27], etc. Effective utilization of the three-dimensional structural information has the potential to increase the performance of PPI sites prediction methods. Graph neural networks (GNN) have been emerged as an effective tool for encoding structural information [43, 44, 45, 46, 47]. However, although GNN based architecture has been applied to pairwise binding site prediction [46, 47], it has not been used for predicting the binding sites of a single protein. Moreover, unlike methods that may not appropriately encode residue-specific information about long-range interactions as they learn a single global feature representation for all the residues (e.g., global representation of proteins by [15]), GNNs have the potential to effectively encode global features involving any specific residue, by learning a suitable function for *a particular residue and its close proximity neighbours*.

Among various GNN based architectures, Graph Attention Network (GAT) was proved to be very effective in protein-interaction network related problem [44]. GAT uses attention mechanism [48, 49] in the node-level aggregation process, which helps it perform a weighted aggregation. But the originally proposed architecture of GAT does not consider the features of the edges, either during *the aggregation process* or during *the calculation of the attention score*. Therefore, GAT lacks the ability to utilize the rich structural information that might have been encoded in the edge-features.

In this paper we present a novel variant of GAT, which we call **E**dge Aggregated **G**raph **At**tention Network (EGAT), for predicting interaction sites of a single (isolated) protein with

known structure. Unlike GAT, EGAT is expected to effectively leverage the structural information encoded in the edge-features during both the aggregation and attention score calculation phases. We also present a successful utilization of transfer-learning from pretrained transformer-like models by Elnaggar *et al.* [36] in PPI sites prediction. Combined with the transfer-learning, our proposed EGAT architecture achieved substantial improvement over the best alternate methods on the widely used benchmark dataset assembled by Zeng *et al.* [15]. EGAT also contributes towards interpretable deep learning models – models that are able to summarize the reasons of the network behavior, or produce insights about the causes of their decisions.

2 Approach

2.1 Feature Representation

2.1.1 Graph representation of proteins

The proposed model EGAT is a graph neural network based architecture, and we represent the three-dimensional structure of each protein P in our dataset as a directed k-nearest neighbor graph G [50]. The set $V(G)$ of nodes in the graph G is the set of the amino-acid residues of a protein P. Let $\mathcal{N}_i$ be *the neighborhood* of node (residue) i comprising its k nearest neighbors (i.e., $|\mathcal{N}_i| = k$), and i be the *center of this neighborhood*. Each node $i \in V(G)$ is connected by directed edges to every node in $\mathcal{N}_i$. These neighbours of i are selected by sorting all other nodes based on their distances from i, and then taking the nearest k nodes, where k is a hyperparameter of our method. Inspired by the success of pair-wise protein-protein interaction prediction by [46], the distance between any two nodes (residues) is calculated by averaging the distances between their atoms (using the atom coordinates from the PDB files [42]). As each residue is represented as a node in our graph representation, we use the terms 'residue' and 'node' interchangeably for convenience.

2.1.2 Node-level feature representation

Each node $i \in V(G)$ (representing the i-the residue in a protein sequence) is represented by a feature vector. EGAT takes as input a sequence $X = \{X_1, X_2, X_3, \ldots, X_N\}$ of amino-acid residues of the protein P, where X_i is the one-letter notation [51] of the i-th residue and N is the total number of residues in P. X is passed through the embedding generation pipeline developed by Elnaggaret *et al.* [36]. Although we have used ProtBERT for our experiments since it was shown to achieve superior performance compared to other methods on residue-level classification tasks (e.g. secondary strucuture prediction), EGAT is agnostic to the pretrained language models available in ProtTrans [36], which means that other appropriate language models in ProtTrans can also be used in our model. ProtBERT generates a sequence of embedding vectors $q = \{q_1, q_2, q_3, \ldots, q_N\}$, $q_i \in \mathbb{R}^{d_{protbert}}$ ($d_{protbert} = 1024$), where q_i is used as the node-level feature vector of node i.

2.1.3 Edge-level feature representation

In the directed graph representation G of a protein P, the edge features of an edge E_{ij} (from node i to node j) in G is denoted by ξ_{ji}, where $\xi_{ji} \in \mathbb{R}^{f_\varepsilon}$, f_ε is the number of features of the edge. We used the following two features (i.e., $f_\varepsilon = 2$) as edge-features: (1) distance D_{ij} between the residues i and j, which is calculated by taking the average distances between their atoms, and (2) relative orientation θ_{ij} of the residues i and j, which is calculated as the absolute value of the angle between the surface-normals of the planes of these two residues that go through the alpha Carbon

atom (C_α), Carbon atom of the Carboxyl group, and Nitrogen atom of the Amino group of each of the residues. We standardize both these features across all the training samples.

2.2 Architecture of EGAT

The architecture of EGAT can be split into three separate discussions: 1) the architecture of local feature extractor, 2) the architecture of our proposed edge aggregated graph attention layer, and finally 3) the node-level classification.

2.2.1 Local feature extractor

A local feature extractor λ is applied, as shown in Figure 1(a), to a graph representation G of an arbitrary protein P. This layer is expected not only to capture the *local interactions* of the residues of the protein (sequence), but also to *reduce the dimension* of the node-level feature-vectors $q = \{q_1, q_2, q_3, \ldots, q_N\}$. This helps the model learn to filter out unnecessary information, keeping the most useful information from q. Also, this helps the model to avoid overfitting, as this reduces the number of parameters for the subsequent layer.

We used a one-dimensional convolutional neural network layer with a window size w_{local} as λ. Here, w_{local} is preferably a relatively small odd number to capture information about the relationship among the residues that are *sequentially closer*, but may or may not be closer in three-dimensional Euclidean space. The motivation behind taking an odd number is to ensure equal number of residues from both sides of a particular residue i. The sequence q of node features is passed through λ to generate a lower dimensional representation $h = \{h_1, h_2, h_3, \ldots, h_N\}$, $h_i \in \mathbb{R}^{f_\eta}$, where $f_\eta < d_{protbert}$. Here, for a residue i, λ encodes the feature-vectors $\{q_j | q_j \in q \ and \ (i - \frac{w_{local}-1}{2}) \leq j \leq (i + \frac{w_{local}-1}{2})]\}$ into a new condensed feature representation h_i for the node i.

2.2.2 Edge aggregated graph attention layer

We now describe the original Graph Attention layer [44] and our proposed modifications by introducing edge aggregations. The feature representations h (generated by the local feature extractor, λ) are transformed using our proposed edge aggregated graph attention layer Υ to encode the three-dimensional structural information of proteins.

In various Graph Neural Network base architectures [43, 44], there is an aggregation process where for a node i the feature representations of all the neighboring nodes $\mathcal{N}_i$ are aggregated to generate a fixed sized new representation U_i for the node i, which is then used for further computations. One highly used aggregation process is the weighted average of the features of the neighboring nodes. In such process, the representation of the i-th node $U_i \in \mathbb{R}^{\hat{f}_\eta}$ is generated according to the following Equation 1.

$$U_i = \sigma(\sum_{j \in \mathcal{N}_i} \gamma_{ji} W^\nu h_j) \tag{1}$$

Here, $\mathcal{N}_i$ is the neighborhood of the node i. Also, h_j is the local feature representation of the node $j \in \mathcal{N}_i$, $W^\nu \in \mathbb{R}^{\hat{f}_\eta \times f_\eta}$ is a learnable parameter, and γ_{ji} is the weight that indicates how much important the features of the node j are to the node i. For Graph Convolutional Networks (GCN) [43] and Const-GAT [44], $\forall j \in \mathcal{N}_i$, $\gamma_{ji} = \frac{1}{|\mathcal{N}_i|} = \frac{1}{k}$, which is a constant. On the other hand, for GAT [44], γ_{ji} is a function $\mathcal{F}(.)$ of the features h_i and h_j of the nodes i and j, respectively, representing the attention score of edge E_{ji} (see Eqn. 2). In 2018, Veličković *et al.* [44] showed the

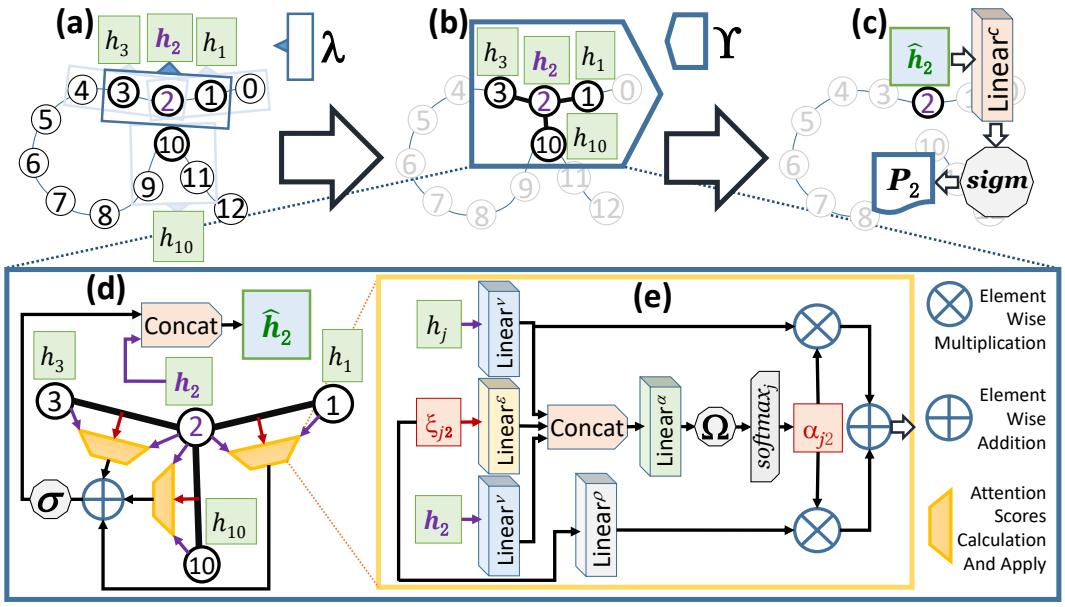


Figure 1: **Schematic diagram of the overall pipeline of EGAT being applied to a dummy protein having 13 residues ($i = 0, 1, 2, \ldots, 12$).** (a) Local feature extractor λ (with window size $w_{local} = 3$) being applied. (b) Application of the edge aggregated graph attention layer Υ to a neighborhood where residue 2 is the center node, and $\{1, 3, 10\} \in \mathcal{N}_2$ (neighborhood of node 2). (c) Node-level classifier applied on the final representation $\hat{h}_2$ of node 2 (generated by Υ). Here *sigm* represents the sigmoid activation function to generate a probability P_2, which represents the numeric propensity of node 2 for being an interaction site. (d) The details on the edge aggregated graph attention layer Υ shown in an expanded for (expansion is shown using dotted lines). Here, the yellow trapezoids represent the modules that calculate the attention scores and apply them to the features of the nodes and the edges. (e) The underlying working mechanism of a yellow trapezoid is shown in details in an expanded form. Here, h_j represents the feature representation of the node $j \in \mathcal{N}_2$, ξ_{j2} represents the feature vector of the edge from node j to node 2. The $softmax_j$ represents the softmax normalization applied to generate a normalized attention score α_{j2} for the edge from node j to node 2. In this figure, σ and Ω represent two activation functions, and **Linear**^{x} ($x \in \{c, \nu, \varepsilon, \rho, \alpha\}$) represents a linear layer with learnable parameter W^x.

significant impact of these attention scores on the protein-protein interaction (PPI) dataset [52] consisting of graphs corresponding to different human tissues.

$$\gamma_{ji} = \mathcal{F}(h_i, h_j) \tag{2}$$

Using edge features during the calculation of the attention scores. In the calculation of the attention score as in Eqn. 2, in addition to node features $h_i, h_j \in \mathbb{R}^{f_\eta}$, we incorporate ξ_{ji}, the edge features of the directed edge from node j to node i. Equations 3 and 4 show the computations to generate the attention scores that are dependant not only on the nodes but also on the edges. Equation 3 represents a scoring function that is parameterized by a learnable parameter $W^\alpha \in \mathbb{R}^{2\hat{f}_\eta + \hat{f}_\varepsilon}$, and $\Omega(.)$ is an activation function.

$$e_{ji} = \Omega(W^\alpha [W^\nu h_i || W^\nu h_j || W^\rho \xi_{ji}]) \tag{3}$$

In Eqn. 3, e_{ji} is an unnormalized representation of the attention score and the symbol "||" represents the concatenation operation. Here, $W^{\nu} \in \mathbb{R}^{\hat{f}_{\eta} \times f_{\eta}}$ and $W^{\rho} \in \mathbb{R}^{\hat{f}_{\varepsilon} \times f_{\varepsilon}}$ are learnable parameters used to apply linear transformation on the features of the nodes and the edges respectively.

$$\alpha_{ji} = softmax_j(e_{ji}) = \frac{exp(e_{ji})}{\sum_{k \in \mathcal{N}_i} exp(e_{ki})} \tag{4}$$

Equation 4 computes a softmax normalization (using the function $softmax_j(.)$) on $\{e_{ji} | j \in \mathcal{N}_i\}$ following Bahdanau *et al.* [48]. This gives us a probability distribution over all the nodes $j \in \mathcal{N}_i$ (i being the center node) which is the attention distribution $\{\alpha_{ji} | j \in \mathcal{N}_i\}$.

Using edge features during the aggregation process. In order to utilize the full potential of the edge features ξ_{ji}, we propose to aggregate them along side the features of the neighboring nodes $\{h_j | j \in \mathcal{N}_i\}$, where node i is at the center of the neighborhood. We have updated Eqn. 1 accordingly and come up with Eqn. 5. Here, $\hat{h}_i \in \mathbb{R}^{\hat{f}_{\eta}}$ is the final feature representation of the node i after incorporating our proposed edge aggregated graph attention layer Υ.

$$\hat{h}_i = \sigma(\sum_{j \in \mathcal{N}_i} \alpha_{ji} W^{\nu} h_j + \sum_{j \in \mathcal{N}_i} \alpha_{ji} W^{\varepsilon} \xi_{ji}) \; || \; h_i \tag{5}$$

Here, W^{η} is the same learnable parameter as in Equation 3, and $W^{\varepsilon} \in \mathbb{R}^{\hat{f}_{\eta} \times f_{\varepsilon}}$ is a new learnable parameter, which is multiplied with the edge feature vector ξ_{ji} to apply a linear transformation on ξ_{ji} before aggregation. Here, $\sigma(.)$ is an activation function and the output of $\sigma(.)$ is concatenated with h_i (the previous feature representation of the node i) and thereby generating the new representation $\hat{h}_i$, which is the output of our edge aggregated graph attention layer Υ. Figure 1(b) demonstrates Υ being applied on a neighborhood of a dummy protein. Figures 1(d) and 1(e) show the detailed mechanism of Υ layer.

2.2.3 Node-level classification

For each node $i \in V(G)$, the final feature representation $\hat{h}_i$ (generated by edge aggregated graph attention layer Υ) is linearly transformed to produce a single scalar followed by an application of the sigmoid activation function as shown in Eqn. 6. This generates a probability $\hat{P}_i \in [0, 1]$, representing the numeric propensity $\hat{P}_i$ of residue i for interactions with other proteins. Here, $W^c \in \mathbb{R}^{1 \times \hat{f}_{\eta}}$ is a learnable parameter and $sigm(.)$ is the Sigmoid activation function [53]. Figure 1(c) shows the node-level classification in EGAT.

$$P_i = sigm(W^c \hat{h}_i) \tag{6}$$

2.3 Overall end-to-end pipeline of EGAT

Figure 1 shows the overall end-to-end pipeline of EGAT. Here we demonstrate EGAT being applied on a dummy protein with thirteen residues and show the computation for only residue 2 for the sake of readablity and clarity of this figure. EGAT starts with representing a protein as a graph G, where each node $i \in V(G)$ is connected to $|\mathcal{N}_i| = k$ other closest nodes with directed edges. For the sake of readability, we used $|\mathcal{N}_i| = k = 3$ in this example. Here $\mathcal{N}_2 = \{1, 3, 10\}$. Not that residue 10 is not sequentially closer to node 2, but is in close proximity of node 2 in three-dimensional space. See Appendix A for more details.

2.4 GAT-PPI: GAT based PPI prediction without edge aggregation

As this is the first known study on leveraging graph neural networks for PPI prediction for single proteins, we have developed an original GAT [44] based PPI prediction approach without any edge aggregation in order to show the superiority of graph based approach over other competing approaches. We call this method GAT-PPI. We used the original implementation provided by the Deep Graph Library [54].

3 Results and Discussion

3.1 Dataset

We analyzed three widely used benchmark datasets, namely (1) Dset_186 [25], (2) Dset_72 [25], and (3) PDBset_164 [29]. Dset_186, Dset_72, and PDBset_164 contain 186, 72, and 164 non-repetitive protein sequences, respectively. An amino acid is labeled as an interaction site if its absolute solvent accessibility is less than 1 $Å^2$, before and after the binding of protein in the binding form; otherwise, it is labeled as a non-interaction site [15, 22, 55]. All these three datasets have been built with proteins from PDB-database [42], with sequence homology less than 25% and resolution less than 3.0 Å (solved by X-ray crystallography). As these datasets come from different research groups, [15] integrated the three datasets into a fused dataset to ensure that the training set and the test set are from an identical distribution. See Appendix B for more details.

3.2 Evaluation metrics

For the evaluation of EGAT, we used seven widely used evaluation metrics [15, 20, 21], namely accuracy, precision, recall, F1-measure (F1), area under the receiver operating characteristic curve (AUROC), area under the precision-recall curve (AUPRC), Matthews correlation coefficient (MCC). See Appendix C for more details. As rightly mentioned by [20], AUROC and AUPR convey a comprehensive performance measurement of a method since these two metrics are threshold independent. Among the other metrics, F1-measure and MCC are the most important performance metrics since PPISP is an imbalanced learning problem [15, 56, 57]. We performed Wilcoxon signed-rank test [58] (with $\alpha = 0.05$) to measure the statistical significance of the differences between two methods.

3.3 Methods

We compared our proposed EGAT and GAT-PPI (the proposed model without edge aggregation) with nine other competing methods for predicting PPI sites, namely SPPIDER [27], ISIS [28], PSIVER [25], SPRINGS [29], RF_PPI [18], and especially the most recent and the most accurate predictors IntPred [16], SCRIBER [21], DeepPPISP [15] and DELPHI [20]. Among these nine alternate methods, SPIDER, IntPred and DeepPPISP are strucuture-based methods that leverage structural data or features derived from structural data. See Appendix D for further details.

3.4 Results on benchmark dataset

The comparison of EGAT with other state-of-the-art methods are shown in Table1. Remarkably, EGAT outperformed all other methods under six (out of seven) evaluation metrics, including the most important ones (e.g., AUROC, AUPRC, MCC, F1). Recall is the only metric where EGAT was outperformed by GAT-PPI (which is our proposed model without edge aggregation). Notably, GAT-PPI also achieved the second best performance on the four important evaluation metrics,

namely F1, MCC, AUROC, AUPRC, which are second to only EGAT. Thus, our proposed GAT based models EGAT (with edge aggregation) and GAT-PPI (without edge aggregation) are the best and second-best methods, respectively on this benchmark dataset. These results clearly show the superiority of the proposed GAT based architecture (with or without edge aggregation) over the best alternate methods. The AUPRC and MCC – two of the most important evaluation metrics due to the imbalance nature of PPI prediction problem – obtained by EGAT are 0.438, 0.405 and 0.27, respectively, which are 4.8%, 12.5% and 14.4% higher than those achieved by the best existing method DELPHI, and these improvements are statistically significant (p-value < 0.05). Table A3 in Appendix E shows the statistical significance analyses of EGAT, GAT-PPI and DELPHI for the four most important evaluation metrics.

Table 1: A comparison of the predictive performance of our proposed EGAT and GAT-PPI with other state-of-the-art methods on the benchmark dataset. The best and the second best results for each metric are shown in bold and italic, respectively. Values which were not reported by the corresponding source are indicated by "-".

Method	ACC	Precision	Recall	F1	AUROC	AUPRC	MCC
SPPIDER[1,2]	0.622	0.209	0.459	0.287	-	0.23	0.089
ISIS[2]	*0.694*	0.211	0.362	0.267	-	0.24	0.097
PSIVER[2]	0.653	0.253	0.468	0.328	-	0.25	0.138
SPRINGS[2]	0.631	0.248	0.598	0.35	-	0.28	0.181
RF_PPI[2]	0.598	0.173	0.512	0.258	-	0.21	0.118
IntPred[1,2]	0.672	0.247	0.508	0.332	-	-	0.165
SCRIBER	0.616	0.274	0.569	0.37	0.635	0.307	0.159
DeepPPISP[1,2]	0.655	0.303	0.577	0.397	0.671	0.32	0.206
DELPHI	0.667	*0.32*	*0.604*	0.418	0.69	0.36	0.236
GAT-PPI[1]	0.653	0.318	**0.659**	*0.429*	*0.714*	*0.398*	*0.252*
EGAT[1]	**0.715**	**0.358**	0.561	**0.438**	**0.719**	**0.405**	**0.27**

[1] Uses structural information.
[2] Results reported by DeepPPISP [15].

Notably, our proposed GAT based model GAT-PPI, even without the edge aggregation, outperformed other existing methods including DELPHI and DeepPPISP, and the improvements are statistically significant. EGAT is substantially better than two of the most recent structure-based methods, namely DeepPPISP and IntPred. EGAT achieved 9.1%, 9.2%, 10.3%, 7.2%, 26.6% and 31.1% higher scores for the above-mentioned metrics respectively than those of DeepPPISP. In particular, 26.6% and 31.1% improvement over DeepPPISP in two fo the most important metrics AUPRC and MCC is quite remarkable.

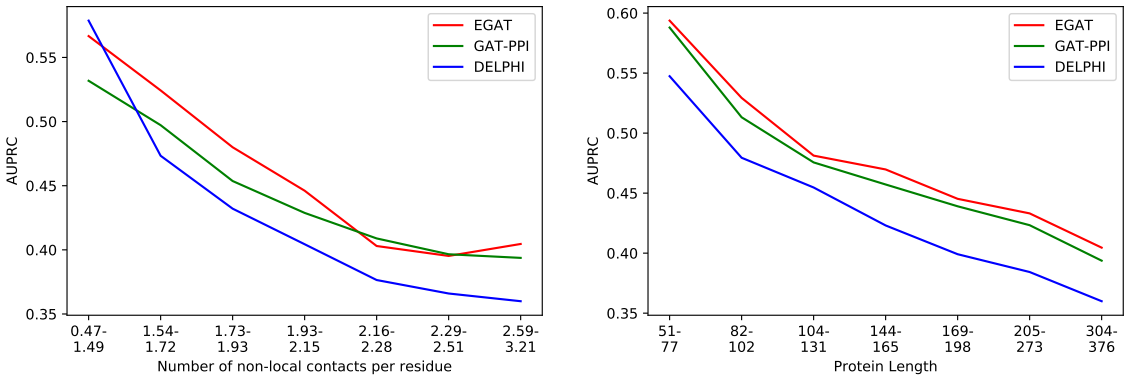
In order to visually show the predicted interfacial sites, we show in Fig. A1 (see Appendix F) the true and predicted (by EGAT and DELPHI) interaction sites on a representative protein (PDB ID 3OUR, Chain-B). This protein is 150 residue long with 39 interaction sites. It has 412 non-local contact pairs, suggesting a high level of long-range interactions.

3.5 Impact of long-range interactions in PPI sites prediction

EGAT, unlike DeepPPISP, is designed for generating different suitable global features for different residues. Therefore, we investigated the performance of EGAT under various levels of long-range interactions, and compared with Delphi and GAT-PPI. We could not include DeepPPISP in this experiments as its protein-wise predictions are not publicly available and the webserver at `http:`

//bioinformatics.csu.edu.cn/PPISP/ is not accessible (last accessed on Oct 10, 2020). The predictions of DELPHI were obtained from DELPHI webserver (available at: https://delphi.csd.uwo.ca/).

We computed the number of non-local interactions per residue for each of the 70 proteins in our test set, and sorted them in an ascending order. Two residues at sequence position i and j are considered to have non-local interaction if they are at least 20 residues apart ($|i - j| \geq 20$), but $< 8\mathring{A}$ away in terms of their atomic distances between $C\alpha$ atoms [59]. Next, we put them in seven equal sized bins $b_1, b_2, \ldots, b_7$ (each containing 10 proteins, where b_1 contains the proteins with the lowest level of non-local interactions (0.41-1.49 non-local contact per residue) and b_7 represents the model condition with the highest level of non-local interactions (2.59-3.21 non-local contact per residue). We show the AUPRC obtained by EGAT, GAT-PPI and DELPHI under various model conditions in Fig. 2 (a) and Table A4 in Appendix G. These results show that – as expected – the performance of both EGAT and DELPHI degrades as we increase the number of non-local contacts. However, the difference in predictive performance between EGAT and DELPHI significantly increases with increasing levels of non-local interactions (with a few exceptions). Note that there is no statistically significant difference between them on b_1 ($p > 0.05$), but as we increase the level of non-local interactions, EGAT and GAT-PPI tend to become more accurate than DELPHI and attain significant improvement on b_7 ($p < 0.05$). These results clearly indicates that addressing non-local interactions by suitable global features is one of the key factors in the improvement.



(a) (b)

Figure 2: **Impact of long-range interactions and protein lengths on predictive performance of PPI sites prediction.** (a) AUPRC of EGAT, GAT-PPI and DELPHI on varying levels of non-local interactions. We show the results on seven bins as shown in Table A4 in Appendix G. (b) AUPRC of EGAT, GAT-PPI and DELPHI on varying lengths of the proteins.

3.6 Impact of protein length

We investigated the impact of protein lengths since we took global features into consideration. We divided 70 proteins in our test set into seven non-overlapping bins based on their lengths. We observed a similar trend as in long-range interactions – the predictive performance deteriorates with increasing lengths of the proteins (see Fig. 2 (b)). EGAT and GAT-PPI consistently outperforms DELPHI across various model conditions and the improvement tend to increase and become statistically significant as we increase the length of the proteins. See Appendix H for more results.

3.7 Impact of edge aggregation in graph attention network

The initial enthusiasm for using edge features was to assist the network in generating an embedding with *richer structural information* for each of the nodes in the graph. Indeed, edge aggregation has a significant impact on PPI sites prediction as supported by the experimental results shown in Table 1. EGAT achieved better performance metrics than GAT-PPI (except for the recall). It obtained 2.1%, 12.6%, 2.1%, 0.7%, 1.8%, 7.1% performance improvement over GAT-PPI in accuracy, F1-score, Precision, AUROC, AUPRC, and MCC, respectively.

3.8 Impact of transfer learning using ProtBERT-based features

We investigated the efficacy of the embeddings of the nodes (residues) generated by ProtBERT compared to other types of feature representations that have been widely used in the PPI literature. We compared the impact of *DeepPPISP Features* containing PSSM, raw sequence features, and eight-state secondary structure features with ProtBERT-based embeddings. We trained EGAT and GAT-PPI using both these feature sets (ProtBERT-based features and DeepPPISP features) and analyzed their predictive performance (see Table 2). The results suggest that both GAT-PPI and EGAT obtained better perfomance with the ProtBERT-based features than those achieved with DeepPPISP features. Notably, EGAT consistently outperformed GAT on both feature sets – suggesting the positive impact of edge aggregation regardless of the choice of feature sets. Moreover, even without the ProtTrans features (i.e., with the DeepPPISP feature set), EGAT is better than or as good as DeepPPISP and DELPHI (See Tables 1 and 2).

Table 2: Impact of different types of features. We show the performance of EGAT and GAT-PPI on DeepPPISP feature set and ProtBERT-based feature set. Best results are shown in bold.

Feature-set	Method	ACC	F-measure	AUROC	AUPRC	MCC
DeepPPISP Features	GAT-PPI	0.658	0.413	0.685	0.336	0.229
	EGAT	0.682	0.418	0.695	0.371	0.238
ProtBERT-based Features	GAT-PPI	0.653	0.429	0.714	0.398	0.252
	EGAT	**0.715**	**0.438**	**0.719**	**0.405**	**0.27**

3.9 Interpretability of the edges and the attention scores

Let the EGAT predicted numeric propensity of interaction for any residue r be $P_r \in [0, 1]$, and the true and predicted labels of r be Y_r and $\hat{Y}_r$, respectively. The joint predicted probability of two residues i and j is $P_i * P_j$. Let $\mathcal{G}_{prot} = \{G_1, G_2, \ldots, G_N\}$ be the set of N graphs representing N proteins in our testset, and $V(\mathcal{G}_{prot})$ and $E(\mathcal{G}_{prot})$ represent the sets of nodes and edges in $\mathcal{G}_{prot}$, respectively. In the following analyses, $(a_i), i \in V(\mathcal{G}_{prot})$, represents an ordered list (sequence) that is ordered by the value of i, where a_i may represent Y_i, $\hat{Y}_i$ or P_i.

In order to assess the interpretability of the edges and its features, we investigated the correlation betweene numeric propensities corresponding to the source and destination nodes of the directed edges in our proposed graph based model. More specifically, for an edge $\{E_{ij} | E_{ij} \in E(\mathcal{G}_{prot})\ and\ i, j \in V(\mathcal{G}_{prot})\}$, we computed the correlation coefficient between the two ordered lists (P_i) and (P_j). We ran this analysis on the entire test set using the pearson correlation function implemented in the library Scikit-Learn. We found that there is a high positive correlation between them (correlation coefficient $r = 0.782$), and this correlation is statistically significant with p-value $= 0.0$. This shows us that two nodes are likely to be predicted as the same type (either interaction

or non-interaction site) by EGAT, if there is an edge between them. We also found that for a particular interaction site, its neighbors with relatively higher attention scores are more likely to be an interaction site than its neighbors with relatively lower attention scores (see Appendix I).

3.10 Running time

EGAT is much more faster than the best alternate structure-based method DeepPPISP. We report feature generation time on the smallest (39 amino acids) and the largest (500 amino acids) protein sequences in the test set (see Table A8 in Appendix J). These results suggest that generating ProtBERT-based features is remarkably faster than generating the DeepPPISP features. For example, generating ProtBERT-based features took around only one minute for a 500 amino acid long protein, whereas it took around 2.5 hours for the DeepPPISP features since PSSM generation is very time-consuming. Moreover, PPI prediction time of EGAT, given the generated features, is also faster than DeepPPISP.

4 Conclusions

We have presented EGAT, a novel, highly accurate, and fast method for PPISP for isolated proteins. We have augmented GAT with edge aggregation and demonstrated its efficacy in improving the performance of PPISP. We also, for the first time, utilized transfer-learning with ProtBERT generated feature sets in PPISP. Our experimental results suggest that GAT (with or without edge aggregation) is substantially better than other competing methods. We systematically analyzed the effects of our proposed edge aggregation and transfer learning with pretrained transformer-like models, and revealed that both of them have positive impact on PPISP. Furthermore, we investigated the performance of different methods under various model conditions with varying levels of long-range interactions and protein lengths, and demonstrated the superiority of our proposed methods (especially in harder model conditions). We also demonstrated how this EGAT model can contribute towards improving the interpretability of deep neural network based approaches. See Appendix K for additional discussion.

References

[1] Javier De Las Rivas and Celia Fontanillo. Protein–protein interactions essentials: key concepts to building and analyzing interactome networks. *PLoS Comput Biol*, 6(6):e1000807, 2010.

[2] Naoki Orii and Madhavi K Ganapathiraju. Wiki-pi: a web-server of annotated human protein-protein interactions to aid in discovery of protein function. *PloS one*, 7(11):e49029, 2012.

[3] Khaled S Ahmed, Nahed H Saloma, and Yasser M Kadah. Improving the prediction of yeast protein function using weighted protein-protein interactions. *Theoretical Biology and Medical Modelling*, 8(1):11, 2011.

[4] Xingyi Li, Wenkai Li, Min Zeng, Ruiqing Zheng, and Min Li. Network-based methods for predicting essential genes or proteins: a survey. *Briefings in bioinformatics*, 21(2):566–583, 2020.

[5] Uros Kuzmanov and Andrew Emili. Protein-protein interaction networks: probing disease mechanisms using model systems. *Genome medicine*, 5(4):1–12, 2013.

[6] Rod K Nibbe, Salim A Chowdhury, Mehmet Koyutürk, Rob Ewing, and Mark R Chance. Protein–protein interaction networks and subnetworks in the biology of disease. *Wiley Interdisciplinary Reviews: Systems Biology and Medicine*, 3(3):357–367, 2011.

[7] Ioanna Petta, Sam Lievens, Claude Libert, Jan Tavernier, and Karolien De Bosscher. Modulation of protein–protein interactions for the development of novel therapeutics. *Molecular Therapy*, 24(4):707–718, 2016.

[8] Olivier Sperandio. Toward the design of drugs on protein-protein interactions. *Current pharmaceutical design*, 18(30):4585, 2012.

[9] Shoshana J Wodak, James Vlasblom, Andrei L Turinsky, and Shuye Pu. Protein–protein interaction networks: the puzzling riches. *Current opinion in structural biology*, 23(6):941–953, 2013.

[10] Leandra M Brettner and Joanna Masel. Protein stickiness, rather than number of functional protein-protein interactions, predicts expression noise and plasticity in yeast. *BMC systems biology*, 6(1):128, 2012.

[11] AA Terentiev, NT Moldogazieva, and KV Shaitan. Dynamic proteomics in modeling of the living cell. protein-protein interactions. *Biochemistry (Moscow)*, 74(13):1586–1607, 2009.

[12] Tobias Hamp and Burkhard Rost. More challenges for machine-learning protein interactions. *Bioinformatics*, 31(10):1521–1525, 2015.

[13] Iakes Ezkurdia, Lisa Bartoli, Piero Fariselli, Rita Casadio, Alfonso Valencia, and Michael L Tress. Progress and challenges in predicting protein–protein interaction sites. *Briefings in bioinformatics*, 10(3):233–246, 2009.

[14] Loic Giot, Joel S Bader, C Brouwer, Amitabha Chaudhuri, Bing Kuang, Y Li, YL Hao, CE Ooi, Brian Godwin, E Vitols, et al. A protein interaction map of drosophila melanogaster. *science*, 302(5651):1727–1736, 2003.

[15] Min Zeng, Fuhao Zhang, Fang-Xiang Wu, Yaohang Li, Jianxin Wang, and Min Li. Protein–protein interaction site prediction through combining local and global features with deep neural networks. *Bioinformatics*, 36(4):1114–1120, 2020.

[16] Thomas C Northey, Anja Barešić, and Andrew CR Martin. Intpred: a structure-based predictor of protein–protein interaction sites. *Bioinformatics*, 34(2):223–229, 2018.

[17] Tristan T Aumentado-Armstrong, Bogdan Istrate, and Robert A Murgita. Algorithmic approaches to protein-protein interaction site prediction. *Algorithms for Molecular Biology*, 10(1):7, 2015.

[18] Qingzhen Hou, Paul FG De Geest, Wim F Vranken, Jaap Heringa, and K Anton Feenstra. Seeing the trees through the forest: sequence-based homo-and heteromeric protein-protein interaction sites prediction using random forest. *Bioinformatics*, 33(10):1479–1487, 2017.

[19] Juan Fernandez-Recio, Maxim Totrov, and Ruben Abagyan. Identification of protein–protein interaction sites from docking energy landscapes. *Journal of molecular biology*, 335(3):843–865, 2004.

[20] Yiwei Li, G Brian Golding, and Lucian Ilie. DELPHI: accurate deep ensemble model for protein interaction sites prediction. *Bioinformatics*, 08 2020. btaa750.

[21] Jian Zhang and Lukasz Kurgan. Scriber: accurate and partner type-specific prediction of protein-binding residues from proteins sequences. *Bioinformatics*, 35(14):i343–i353, 2019.

[22] Susan Jones and Janet M Thornton. Analysis of protein-protein interaction sites using surface patches. *Journal of molecular biology*, 272(1):121–132, 1997.

[23] Raphael Townshend, Rishi Bedi, Patricia Suriana, and Ron Dror. End-to-end learning on 3d protein structure for interface prediction. In *Advances in Neural Information Processing Systems*, pages 15642–15651, 2019.

[24] Shandar Ahmad and Kenji Mizuguchi. Partner-aware prediction of interacting residues in protein-protein complexes from sequence data. *PLoS One*, 6(12):e29104, 2011.

[25] Yoichi Murakami and Kenji Mizuguchi. Applying the naïve bayes classifier with kernel density estimation to the prediction of protein–protein interaction sites. *Bioinformatics*, 26(15):1841–1848, 2010.

[26] Zhi-Sen Wei, Ke Han, Jing-Yu Yang, Hong-Bin Shen, and Dong-Jun Yu. Protein–protein interaction sites prediction by ensembling svm and sample-weighted random forests. *Neurocomputing*, 193:201–212, 2016.

[27] Aleksey Porollo and Jarosław Meller. Prediction-based fingerprints of protein–protein interactions. *Proteins: Structure, Function, and Bioinformatics*, 66(3):630–645, 2007.

[28] Yanay Ofran and Burkhard Rost. Isis: interaction sites identified from sequence. *Bioinformatics*, 23(2):e13–e16, 2007.

[29] G Singh, K Dhole, PP Pai, and S Mondal. Springs: Prediction of protein-protein interaction sites using artificial neural networks. *J Proteomics Computational Biol*, 1(1):7, 2014.

[30] Changhui Yan, Drena Dobbs, and Vasant Honavar. A two-stage classifier for identification of protein–protein interface residues. *Bioinformatics*, 20(suppl_1):i371–i378, 2004.

[31] Xiaoying Wang, Bin Yu, Anjun Ma, Cheng Chen, Bingqiang Liu, and Qin Ma. Protein–protein interaction sites prediction by ensemble random forests with synthetic minority oversampling technique. *Bioinformatics*, 35(14):2395–2402, 2019.

[32] Josip Mihel, Mile Šikić, Sanja Tomić, Branko Jeren, and Kristian Vlahoviček. Psaia–protein structure and interaction analyzer. *BMC structural biology*, 8(1):21, 2008.

[33] MR Uddin, S Mahbub, MS Rahman, and MS Bayzid. Saint: Self-attention augmented inception-inside-inception network improves protein secondary structure prediction. *Bioinformatics (Oxford, England)*, 2020.

[34] Jack Hanson, Kuldip Paliwal, Thomas Litfin, Yuedong Yang, and Yaoqi Zhou. Improving prediction of protein secondary structure, backbone angles, solvent accessibility and contact numbers by using predicted contact maps and an ensemble of recurrent and residual convolutional neural networks. *Bioinformatics*, 35(14):2403–2410, 2019.

[35] Stephen F Altschul, Thomas L Madden, Alejandro A Schäffer, Jinghui Zhang, Zheng Zhang, Webb Miller, and David J Lipman. Gapped blast and psi-blast: a new generation of protein database search programs. *Nucleic acids research*, 25(17):3389–3402, 1997.

[36] Ahmed Elnaggar, Michael Heinzinger, Christian Dallago, Ghalia Rihawi, Yu Wang, Llion Jones, Tom Gibbs, Tamas Feher, Christoph Angerer, Debsindhu Bhowmik, et al. Prottrans: Towards cracking the language of life's code through self-supervised deep learning and high performance computing. *arXiv preprint arXiv:2007.06225*, 2020.

[37] Zihang Dai, Zhilin Yang, Yiming Yang, Jaime G Carbonell, Quoc Le, and Ruslan Salakhutdinov. Transformer-xl: Attentive language models beyond a fixed-length context. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2978–2988, 2019.

[38] Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Russ R Salakhutdinov, and Quoc V Le. Xlnet: Generalized autoregressive pretraining for language understanding. In *Advances in neural information processing systems*, pages 5753–5763, 2019.

[39] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics.

[40] Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. Albert: A lite bert for self-supervised learning of language representations. In *International Conference on Learning Representations*, 2019.

[41] Jesse Vig, Ali Madani, Lav R Varshney, Caiming Xiong, Richard Socher, and Nazneen Fatema Rajani. Bertology meets biology: Interpreting attention in protein language models. *arXiv preprint arXiv:2006.15222*, 2020.

[42] Helen M Berman, John Westbrook, Zukang Feng, Gary Gilliland, Talapady N Bhat, Helge Weissig, Ilya N Shindyalov, and Philip E Bourne. The protein data bank. *Nucleic acids research*, 28(1):235–242, 2000.

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

[44] Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. Graph attention networks. In *International Conference on Learning Representations*, 2018.

[45] Yue Wang, Yongbin Sun, Ziwei Liu, Sanjay E Sarma, Michael M Bronstein, and Justin M Solomon. Dynamic graph cnn for learning on point clouds. *Acm Transactions On Graphics (tog)*, 38(5):1–12, 2019.

[46] Alex Fout, Jonathon Byrd, Basir Shariat, and Asa Ben-Hur. Protein interface prediction using graph convolutional networks. In *Advances in neural information processing systems*, pages 6530–6539, 2017.

[47] Yi Liu, Hao Yuan, Lei Cai, and Shuiwang Ji. Deep learning of high-order interactions for protein interface prediction. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 679–687, 2020.

[48] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate. In *3rd International Conference on Learning Representations, ICLR 2015*, 2015.

[49] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008, 2017.

[50] David Eppstein, Michael S Paterson, and F Frances Yao. On nearest-neighbor graphs. *Discrete & Computational Geometry*, 17(3):263–282, 1997.

[51] IUPAC-IUB Tentative Rules. A one letter notation for amino acid sequence. *Biochem. J*, 113:1–4, 1969.

[52] Marinka Zitnik and Jure Leskovec. Predicting multicellular function through multi-layer tissue networks. *Bioinformatics*, 33(14):i190–i198, 2017.

[53] Jun Han and Claudio Moraga. The influence of the sigmoid function parameters on the speed of backpropagation learning. In *International Workshop on Artificial Neural Networks*, pages 195–201. Springer, 1995.

[54] Minjie Wang, Lingfan Yu, Da Zheng, Quan Gan, Yu Gai, Zihao Ye, Mufei Li, Jinjing Zhou, Qi Huang, Chao Ma, et al. Deep graph library: Towards efficient and scalable deep learning on graphs. *arXiv preprint arXiv:1909.01315*, 2019.

[55] Patrick Aloy and Robert B Russell. Interrogating protein interaction networks through structural biology. *Proceedings of the National Academy of Sciences*, 99(9):5896–5901, 2002.

[56] Sjoerd J de Vries and Alexandre MJJ Bonvin. How proteins get in touch: interface prediction in the study of biomolecular complexes. *Current protein and peptide science*, 9(4):394–406, 2008.

[57] Min Zeng, Beiji Zou, Faran Wei, Xiyao Liu, and Lei Wang. Effective prediction of three common diseases by combining smote with tomek links technique for imbalanced medical data. In *2016 IEEE International Conference of Online Analysis and Computing Science (ICOACS)*, pages 225–228. IEEE, 2016.

[58] Frank Wilcoxon, SK Katti, and Roberta A Wilcox. Critical values and probability levels for the wilcoxon rank sum test and the wilcoxon signed rank test. *Selected tables in mathematical statistics*, 1:171–259, 1970.

[59] Rhys Heffernan, Yuedong Yang, Kuldip Paliwal, and Yaoqi Zhou. Capturing non-local interactions by long short-term memory bidirectional recurrent neural networks for improving prediction of protein secondary structure, backbone angles, contact numbers and solvent accessibility. *Bioinformatics*, 33(18):2842–2849, 2017.

[60] Wouter G Touw, Coos Baakman, Jon Black, Tim AH Te Beek, Elmar Krieger, Robbie P Joosten, and Gert Vriend. A series of pdb-related databanks for everyday needs. *Nucleic acids research*, 43(D1):D364–D368, 2015.

[61] Wolfgang Kabsch and Christian Sander. Dictionary of protein secondary structure: pattern recognition of hydrogen-bonded and geometrical features. *Biopolymers: Original Research on Biomolecules*, 22(12):2577–2637, 1983.

[62] Kyunghyun Cho, Bart van Merrienboer, Çaglar Gülçehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. Learning phrase representations using rnn encoder-decoder for statistical machine translation. In *EMNLP*, 2014.

[63] Shandar Ahmad and Akinori Sarai. Pssm-based prediction of dna binding sites in proteins. *BMC bioinformatics*, 6(1):33, 2005.

[64] Alasdair TR Laurie and Richard M Jackson. Q-sitefinder: an energy-based method for the prediction of protein–ligand binding sites. *Bioinformatics*, 21(9):1908–1916, 2005.

Appendix A Overall end-to-end pipeline of EGAT

EGAT converts the residues to a series of tokens (each token representing a residue) and uses ProtBERT [36] to generate an embedding-vector for each of the residues. These embedding-vectors are assigned as the initial feature-representations q of the nodes in $V(G)$. Alongside, the edge features-vectors $\{\xi_{i,j}|i,j\ in\ V(G)\}$ are calculated from the structural data of the protein (available in PDB files [42]). Each features-vector $\xi_{i,j}$ is associated with one directed edge $E_{i,j}$ from node i to node j. Next, local feature extractor λ is applied to the features-representations (q) of the nodes of the proteins. λ generates a new feature representation h_i of a residue i (see Figure 1(a)). The details of local feature extraction have been described in Sec. 2.2.1.

Once the local feature extraction is completed, the edge aggregated graph attention layer Υ is applied on each of the neighborhoods. We show the application of Υ only to the neighborhood of residue 2 in Fig. 1(b)). Please see Sec. 2.2.2 for details. Υ generates the final feature representation $\hat{h}_i$ of the central node i of a neighborhood. Finally, node-level classification (described in details in Sec 2.2.3) is applied to the final representation ($\hat{h}_2$ in this example). Node level classifier provides us with a probability value P_2 for residue 2, which is the predicted propensity of residue 2 being an interfacing residue (or interaction site). This same end-to-end pipeline is applied to all other residues and thereby computing the propensities of all the residues.

Appendix B Datasets

Zeng *et al.* [15] split this fused dataset into a test set comprising 70 randomly selected protein sequences and a training set with the remaining (about 83.4%) protein sequences. They evaluated their method DeepPPISP as well as other state-of-the-art methods on this split using the same evaluation scheme. For the sake of a fair comparison, we used the same splitting and evaluation

scheme used by [15] (available at: `https://github.com/CSUBioGroup/DeepPPISP`). Table A1 shows the numbers of interaction and non-interaction sites in these datasets and the splits used in this study and [15].

We note that DELPHI [20], which is a sequence-based method, used a much larger training dataset containing 9,982 protein sequences. However, leveraging this large dataset for training structure-based methods is difficult due to the unavailability of curated structural information.

Table A1: Summary of the datasets analyzed in this study.

Dataset	Proteins	interaction sites	non-interaction sites
Dset_186	186	1,923 (10.60%)	16,217 (89.40%)
Dset_72	72	5,517 (15.23%)	30,702 (84.77%)
PDBset_164	164	6,096 (18.10%)	27,585 (81.90%)
Train	352	11,079 (15.14%)	62,102 (84.86%)
Test	70	2,332 (19.78%)	9,459 (80.22%)

Appendix C Evaluation metrics

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{7}$$

$$Precision = \frac{TP}{TP + FP} \tag{8}$$

$$Recall = \frac{TP}{TP + FN} \tag{9}$$

$$F1 - measure = \frac{2 * Precision * Recall}{Precision + Recall} \tag{10}$$

$$MCC = \frac{TP * TN - FP * FN}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}} \tag{11}$$

In Eqns. 8-12, true positives (TP) and true negatives (TN) represent the number of correctly predicted interaction and non-interaction sites respectively. Similarly, the number of incorrectly predicted interaction and non-interaction sites are represented by false positives (FP) and false negatives (FN), respectively.

Appendix D Methods

Table A2 shows different types of features used by various methods considered in this study. SPPIDER [27] showed the impact of a combination of putative relative solvent accessibility (RSA) prediction-based fingerprints with other structural and sequence information using Support Vector Machines (SVM) and neural networks. IntPred [16] exploited the random forest algorithm and used various structural features as well as sequence-based features. DeepPPISP [15] is a deep learning based method that used both local and global features generated from eight-state secondary structures (derived from PDB [42] structure files using DSSP [60, 61][1] program), and various sequence-based features (raw protein sequence and position-specific score matrices (PSSM)). DELPHI [20] –

[1] `https://swift.cmbi.umcn.nl/gv/dssp/`

Table A2: Feature sets used by various state-of-the-art methods.

Method	Feature-type	Features
SPPIDER [27]	Sequence- and structure-based	Putative relative solvent accessibility (RSA), Surface patch of protein
IntPred [16]	Sequence- and structure-based	Disulphide bonds, hydrogen bonds and secondary structures (α-helix, β-sheet, mixed secondary structure, and coil), planarity, propensity score, hydrophobicity, homology conservation score and FEP conservation score
DeepPPISP [15]	Sequence- and Structure-based	Eight-state secondary structures, raw protein sequence and position-specific score matrices (PSSM)
DELPHI [20]	Sequence-based	High-scoring segment pair (HSP), a variation of 3-mer amino acid embedding (ProtVec1D), position information, position-specific scoring matrix (PSSM), evolutionary conservation (ECO), putative relative solvent accessibility (RSA), relative amino acid propensity (RAA), putative protein-binding disorder, hydropathy index, physicochemical characteristics, physical properties, and PKx
SCIBBER [21]	Sequence-based	Relative Amino Acid Propensity (RAAP) for binding, Putative Relative Solvent Accessibility (RSA), evolutionary conservation (ECO), Hydrophobicity, Polarity, Charge, protein-binding disordered regions, putative secondary structure, physicochemical properties, relative position, relative position
PSIVER [25]	Sequence-based	Predicted accessibility and PSSM
SPRINGS [29]	Sequence-based	Evolutionary information, averaged cumulative hydropathy and predicted RSA
ISIS [28]	Sequence-based	predicted structural features, evolutionary information
RF_PPI [18]	Sequence-based	Protein size, predicted backbone flexibility, sequence specificity

one the most accurate methods – used an ensemble of two deep learning models (one convolutional based and one bidirectional Gated Recurrent Unit [62] based neural network) as the predictor, and was trained in an ensemble learning fashion using a dataset of 9,982 protein sequences, where each residue is represented by twelve types of features (see Table A2 for more details). SCIBBER [21] used a training set comprising 843 proteins containing several primary sequence-based features and their derivatives (e.g., Relative Amino Acid Propensity (RAAP) for binding, Putative Relative Solvent Accessibility (RSA), Hydrophobicity, etc.), and predicted interacting residues using a two-layer design, where the first and second layers consist of five and one machine learning models, respectively. PSIVER [25] trained a Naïve Bayes classifier on sequence-based features (predicted accessibility and PSSM). SPRINGS [29] proposed a shallow artificial neural network, which uses

various sequence based features (e.g., evolutionary information, averaged cumulative hydropathy and predicted RSA). ISIS [28] used shallow neural networks and combined predicted structural features with evolutionary information as feature representations. RF_PPI [18] presented a study on the evaluation of the importance of various sequence based features using Random Forest (RF) classifier.

Appendix E Statistical significance test

Table A3: The statistical significance of various performance metrics between different pairs of methods. We show the p-values using a Wilcoxon signed rank test.

Method-pair	F-measure	AUROC	AUPRC	MCC
DELPHI, EGAT	0.0042	0.0674	0.0457	0.0079
DELPHI, GAT	0.0153	0.0535	0.0369	0.0372
GAT, EGAT	0.0719	0.1861	0.2139	0.1057

Appendix F Predicted and true interaction sites in a represntative protein

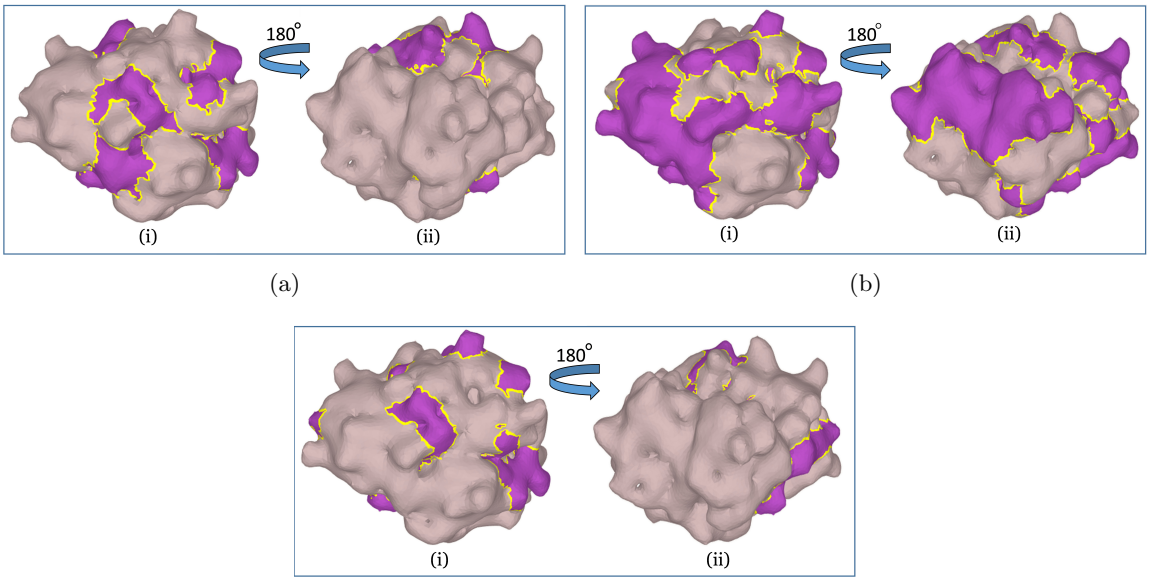


Figure A1: **Interaction sites of a representative protein in the test (PDB ID 3OUR, Chain-B).** (a) Interaction sites predicted by EGAT, (b) interaction sites predicted by DELPHI, and (c) true interactions sites as obtained from the dataset. Interaction sites are shown in purple on the protein surface. The left and right images (i and ii) in each of the figures (a, b and c) show two opposite sides (i.e., 180 rotated view).

Appendix G Impact of long-range interactions

Table A4: AUPRC of EGAT and DELPHI under various levels of non-local interactions. 70 proteins in the testset were divided into 7 disjoint bins each having 10 proteins. The binning was based on the number of non-local contacts per residue in the proteins.

Non-local contact per residue	AUPRC of EGAT	AUPRC of DELPHI	AUPRC Difference	p-value
0.47-1.49	0.5666	0.5787	-0.0121	0.8389
1.54-1.72	0.5243	0.4733	0.0510	0.4062
1.73-1.93	0.4800	0.4321	0.0479	0.2929
1.93-2.15	0.4461	0.4044	0.0417	0.4044
2.16-2.28	0.4030	0.3765	0.0265	0.3515
2.29-2.51	0.3953	0.3660	0.0293	0.1446
2.59-3.21	0.4046	0.3600	0.0446	0.0457

Table A5: AUPRC of GAT-PPI and DELPHI under various levels of non-local interactions. 70 proteins in the testset were divided into 7 disjoint bins each having 10 proteins. The binning was based on the number of non-local contacts per residue in the proteins.

Non-local contact per residue	AUPRC of GAT-PPI	AUPRC of DELPHI	AUPRC Difference	p-value
0.47-1.49	0.5318	0.5787	-0.0469	0.8838
1.54-1.72	0.4972	0.4733	0.0239	0.5218
1.73-1.93	0.4537	0.4321	0.0215	0.4959
1.93-2.15	0.4288	0.4044	0.0244	0.3940
2.16-2.28	0.4089	0.3765	0.0324	0.2764
2.29-2.51	0.3966	0.3660	0.0306	0.1602
2.59-3.21	0.3937	0.3600	0.0338	0.0369

Appendix H Impact of protein length

Table A6: AUPRC of EGAT and DELPHI under various levels of protein lengths. 70 proteins in the testset were divided into 7 disjoint bins each having 10 proteins. The binning was based on the lengths of the proteins.

Protein Length	AUPRC of EGAT	AUPRC of DELPHI	AUPRC Difference	p-value
51-77	0.5937	0.5474	0.0463	0.7539
82-102	0.5293	0.4795	0.0499	0.5796
104-131	0.4813	0.4547	0.0266	0.6856
144-165	0.4697	0.4231	0.0466	0.2726
169-198	0.4453	0.3991	0.0461	0.1204
205-273	0.4331	0.3843	0.0488	0.0374
304-376	0.4046	0.3600	0.0446	0.0457

Table A7: AUPRC of GAT-PPI and DELPHI under various levels of protein lengths. 70 proteins in the testset were divided into 7 disjoint bins each having 10 proteins. The binning was based on the lengths of the proteins.

Protein Length	AUPRC of GAT-PPI	AUPRC of DELPHI	AUPRC Difference	p-value
51-77	0.5879	0.5474	0.0405	0.5391
82-102	0.5132	0.4795	0.0337	0.3506
104-131	0.4756	0.4547	0.0209	0.5287
144-165	0.4572	0.4231	0.0341	0.1666
169-198	0.4390	0.3991	0.0399	0.1056
205-273	0.4233	0.3843	0.0390	0.0503
304-376	0.3937	0.3600	0.0338	0.0369

Appendix I Interpretability of attention scores

In order to investigate the interpretability of the edges as well as to assess the interpretability of the attention scores (edge weights), we investigate – for each residue $i \in V(\mathcal{G}_{prot})$ which is predicted to be an interaction site by EGAT – how much the predictions of its neighbors correlate with their corresponding edge weights (i.e. the attention scores). More specifically, $\forall_i \{i \in V(\mathcal{G}_{prot})\ and\ \hat{Y}_i = 1\}$ we compute the correlation coefficient between the predictions of the neighbors $Y_G = (\hat{Y}_j)_{j \in \mathcal{N}_i}$ and the associated attention scores $A_G = (\alpha_{ji})$. We found a positive correlation coefficient ($r = 0.243$) between $\hat{Y}_G$ and A_G, which is statistically significant (p-value=0.0). This suggests that for a particular interaction site, its neighbors with relatively higher attention scores are more likely to be an interaction site than its neighbors with relatively lower attention scores. We further demonstrate this correlation with a cartoon figure using a representative protein (PDB ID 3OUR, Chain-B), available in the test set (see Fig. A2(a)).

Figure A2(a) shows the neighborhood $\mathcal{N}_{90}$ of residue 90, and Fig. A2(b) shows the weights (attention scores) of the edges in Fig. A2(b) using color gradient. Deeper hues indicate higher levels of attention. This residue is an interaction site and EGAT rightly predicted this. The other interaction sites, predicted by EGAT, in this neighborhood are shown in green. Interestingly, the edges with relatively higher attention scores (darker color hues) are mostly associated with source

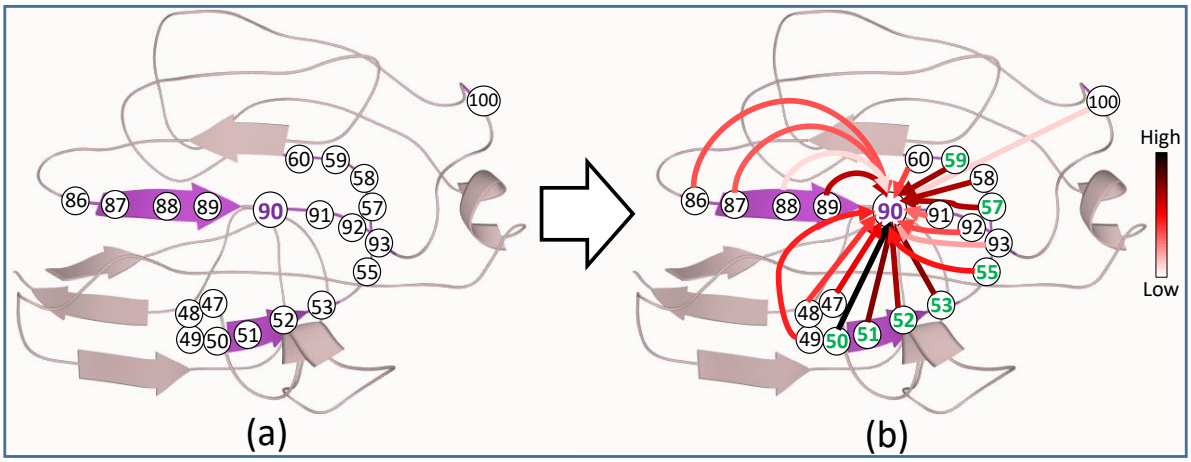


Figure A2: **Interpretability of the edge features produced by EGAT in predicting the interaction sites.** (a) A cartoon images of a protein (PDB ID 3OUR, Chain B), where we show the neighborhood (window size = 20) of the residue 90. (b) The attention scores along the edges in this neighborhood using a color gradient which vary continuously from light red to black with increasing attention scores. The residues shown by green nodes are the interaction sites predicted by EGAT.

nodes that are predicted as interaction sites (residues at positions 50, 51, 52, 53, 55, 57, 59). Among these residues, 50, 52, 53, and 59 are true interaction sites (these four residues are associated to four of the top five attention scores). Moreover, the attention scores of the edges with non-interacting source nodes (e.g., 86, 87, 88, 89, 91, 92, 93) which are closer to 90 in primary sequence are lower than the attention scores of those associated with the long-range interactions (e.g., 50, 51, 52, 53, 59).

While these results are promising, especially considering the black-box nature of other deep learning based methods, they should be interpreted with care. The interaction sites suggested by attention scores *alone* may contain false positives (e.g., residue 51) and false negatives. Higher attention scores do not necessarily guarantee an interaction site, nor is it certain that all the interaction sites within the neighborhood of another interaction site will have relatively higher attention scores. Indeed, the predictions of EGAT does not solely depend on the attention scores as it rightly predicted residue 58 and 89 to be non-interaction sites despite their associate edges have high attention scores. Follow-up studies are required to further investigate the interpretability of such graph based models as well as to design an attention mechanism so that the attention scores are more closely related to true interaction site predictions. This is an interesting research avenue which we left as a future work.

Appendix J Running time

We investigated the time required to generate the features used by EGAT and DeepPPISP (one of most accurate structure-based methods) and the time required for prediction. All analyses were run on the same machine with Intel core i7-7700 CPU (4 cores), 16GB RAM, NVIDIA GeForce GTX 1070 GPU (8GB memory).

Table A8: Running time comparison between EGAT and DeepPPISP. We show the times required for generating the features and performing the prediction on the shortest and longest protein sequences in the test set.

Protein length	Method	Only inference time	Feature extraction time	Total time
500	EGAT	4.54±0.05	63.28±0.1	67.82±0.15
	DeepPPISP	15.02±0.2	8823.07±15	8838.09±15.2
39	EGAT	4.57±0.05	11.84±0.02	16.41±0.07
	DeepPPISP	12.21±0.15	3706.74±6	3718.95±6.15

Appendix K Additional discusssion

Our proposed edge aggregation to effectively capture the structural information is timely considering the growing evidence for the efficacy of structural information in residue-level prediction as well as the increasing availability of structure-known proteins. The demonstrated performance improvement across all seven evaluation metrics is quite promising. Thus, we believe EGAT advances the state-of-the-art in this domain, and will be considered as a useful tool for PPISP. However, this study can be extended in several directions. Training set used by [20] or others is an interesting direction to take, as they contain a large number of training samples compared to the one used in this study. But leveraging these datasets is challenging due to the unavailability of curated structural information. We have adapted and systematically analyzed the ProtBERT model in ProtTrans [36]. Follow-up studies will need to investigate other pretrained language models (e.g., Transformer-XL and XLNet) available in ProtTrans. Although we have focused on the PPI site prediction, EGAT has the potential to be investigated for the prediction of other types of binding sites, including DNA binding sites [63] and ligand (non-protein) binding sites [64]. Another potential research direction is investigating the efficacy of EGAT and GAT-PPI on pair-wise PPI site prediction [23, 46] which we left as a future work.