Bioinformatics
doi.10.1093/bioinformatics/xxxxxx
Advance Access Publication Date: Day Month Year
Manuscript Category

Structural Bioinformatics

EGRET: 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

* 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 EGRET, 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, EGRET offers a more interpretable framework than the typical black-box deep neural networks.
Availability: EGRET is freely available as an open source project at `https://github.com/Sazan-Mahbub/EGRET`.
Keywords: Protein-Protein Interaction Sites, Deep Learning, Graph Neural Network, Edge Aggregation.
Contact: shams_bayzid@cse.buet.ac.bd

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 (De Las Rivas and Fontanillo, 2010). This makes protein-protein interaction (PPI) one of the key elements for understanding underlying biological processes, including functions of the cells (Orii and Ganapathiraju, 2012; Ahmed *et al.*, 2011), PPI networks (Li *et al.*, 2020a; De Las Rivas and Fontanillo, 2010), disease mechanisms (Kuzmanov and Emili, 2013; Nibbe *et al.*, 2011), as well as for designing and developing novel therapeutics (Petta *et al.*, 2016; Sperandio, 2012).

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 (Wodak *et al.*, 2013; Brettner and Masel, 2012; Terentiev *et al.*, 2009). However, the experimental determination of PPI sites is costly and time- and labour-intensive (Hamp and Rost, 2015; Ezkurdia *et al.*, 2009; Giot *et al.*, 2003). 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 (Zeng *et al.*, 2020; Northey *et al.*, 2018; Aumentado-Armstrong *et al.*, 2015; Ezkurdia *et al.*, 2009). These computational methods can be roughly divided into three categories (Hou *et al.*, 2017): (1) Protein-protein docking and modeling (Fernandez-Recio *et al.*, 2004), (2) Structure-based methods (Zeng *et al.*, 2020; Northey *et al.*, 2018; Porollo and Meller, 2007; Chen and Zhou, 2005; La and Kihara, 2012), and (3) Sequence based methods (Li *et al.*, 2020b; Zhang and Kurgan, 2019; Wang *et al.*, 2019b; Hou *et al.*, 2017; Singh *et al.*, 2014; Murakami and Mizuguchi, 2010). Docking and structure based methods, unlike the sequence-based methods, leverage the structural information of the protein molecules.

There are two major areas 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 (Jones and Thornton, 1997). The absence of any information about

the partner proteins makes the latter problem relatively more difficult and challenging (Townshend *et al.*, 2019; Ahmad and Mizuguchi, 2011).

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 (Murakami and Mizuguchi, 2010; Wei *et al.*, 2016; Porollo and Meller, 2007; Hou *et al.*, 2017; Northey *et al.*, 2018; Ofran and Rost, 2007; Singh *et al.*, 2014; Zeng *et al.*, 2020; Li *et al.*, 2020b). Many studies have shown the importance of using local contextual features for predicting interfacing residues (Yan *et al.*, 2004; Zeng *et al.*, 2020; Wang *et al.*, 2019c; Hou *et al.*, 2017; Mihel *et al.*, 2008), 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 (Uddin *et al.*, 2020; Hanson *et al.*, 2019), 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.* (2020) 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 (Zeng *et al.*, 2020), which is also widely used by many state-of-the-art methods (Zeng *et al.*, 2020; Li *et al.*, 2020b; Northey *et al.*, 2018). Other features, such as primary sequence and secondary structure information, alone do not yield good predictions when PSSM features are not used (Zeng *et al.*, 2020). Generation of PSSM using PSI-BLAST (Altschul *et al.*, 1997), however, is a time-consuming process, which Zhang and Kurgan (2019) 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 (Elnaggar *et al.*, 2020). In a very recent study, Elnaggar *et al.* (2020) 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 (Dai *et al.*, 2019) and XLNet (Yang *et al.*, 2019)) and two auto encoding models (BERT (Devlin *et al.*, 2019) and Albert (Lan *et al.*, 2019)) 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 (Devlin *et al.*, 2019)). In a similar study, Vig *et al.* (2020) 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 (Li *et al.*, 2020b; Zeng *et al.*, 2020; Zhang and Kurgan, 2019; Northey *et al.*, 2018; Hou *et al.*, 2017; Singh *et al.*, 2014; Murakami and Mizuguchi, 2010; Ofran and Rost, 2007; Porollo and Meller, 2007; Zhang *et al.*, 2019). However, using only primary sequence based features may limit the capabilities of the methods to achieve higher accuracy (Porollo and Meller, 2007). Thus, several methods have leveraged features derived from structural data (generally from the PDB files (Berman *et al.*, 2000)), including three state and eight state secondary structures (Zeng *et al.*, 2020; Northey *et al.*, 2018), the level of surface exposure to solvent (Porollo and Meller, 2007; Chen and Zhou, 2005), local surface region in a query protein structure (La and Kihara, 2012), 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 (Kipf and Welling, 2017; Veličković *et al.*, 2018; Wang *et al.*, 2019d; Fout *et al.*, 2017; Liu *et al.*, 2020). However, although GNN based architecture has been applied to pairwise binding site prediction (Fout *et al.*, 2017; Liu *et al.*, 2020), 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 Zeng *et al.* (2020)), 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 (Veličković *et al.*, 2018). GAT uses attention mechanism (Bahdanau *et al.*, 2015; Vaswani *et al.*, 2017) 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 **Gr**aph Attention N**et**work (EGRET), for predicting interaction sites of a single (isolated) protein with known structure. Unlike GAT, EGRET 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.* (2020) in PPI sites prediction. Combined with the transfer-learning, our proposed EGRET architecture achieved substantial improvement over the best alternate methods on the widely used benchmark dataset assembled by Zeng *et al.* (2020). EGRET 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 EGRET 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 (Eppstein *et al.*, 1997). The set $V(G)$ of the 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 the node (residue) $i \in V(G)$ comprising its k nearest neighbors (i.e., $|\mathcal{N}_i| = k$), and i be the *center of this neighborhood*. Here i is connected by directed edges to all the nodes 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 Fout *et al.* (2017), the distance between any two nodes (residues) is calculated by averaging the distances between their atoms (using the atom coordinates from the PDB files (Berman *et al.*, 2000)). 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. EGRET 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 (Rules, 1969) 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 Elnaggar *et al.* (2020). 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), EGRET is agnostic to the pretrained language models available in ProtTrans (Elnaggar *et al.*, 2020), which means that other appropriate language models in ProtTrans can also be used in our method. 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_{ji} (from node j to node i) in G is denoted by ξ_{ji}, where $\xi_{ji} \in \mathbb{R}^{f_\varepsilon}$ and 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 of the 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 EGRET

The architecture of EGRET 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 (Veličković *et al.*, 2018) 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 (Kipf and Welling, 2017; Veličković *et al.*, 2018), 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) (Kipf and Welling, 2017) and Const-GAT (Veličković *et al.*, 2018), $\forall j \in \mathcal{N}_i$, $\gamma_{ji} = \frac{1}{|\mathcal{N}_i|} = \frac{1}{k}$, which is a constant. On the other hand, for GAT (Veličković *et al.*, 2018), 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.* (Veličković *et al.*, 2018) showed the significant impact of these attention scores on the protein-protein interaction (PPI) dataset (Zitnik and Leskovec, 2017) 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.* (2015). 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

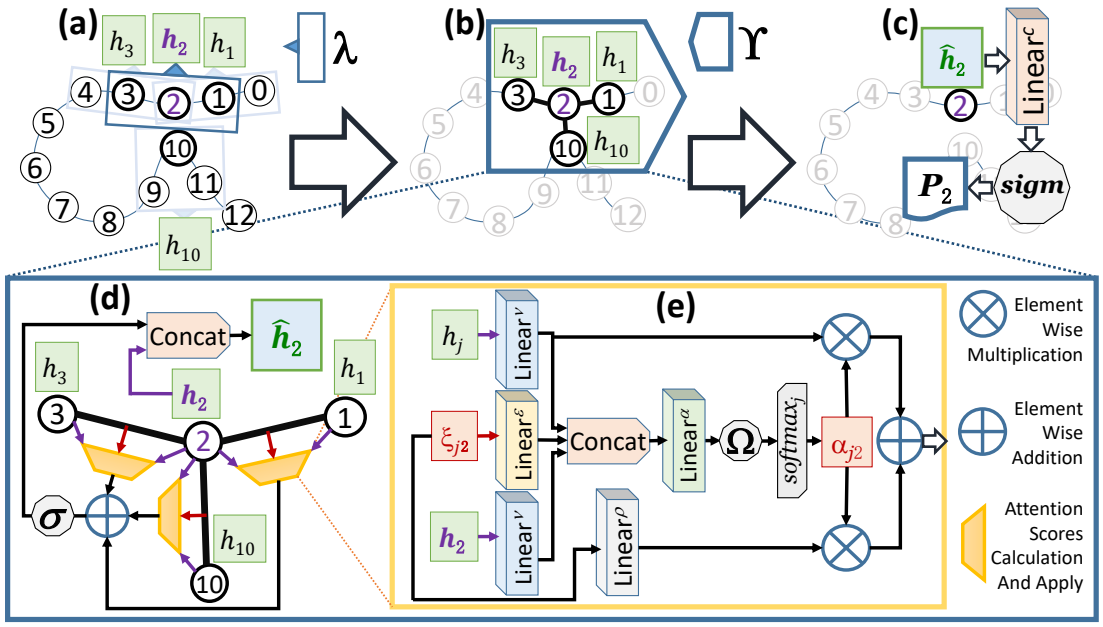

Fig. 1. Schematic diagram of the overall pipeline of EGRET being applied to a dummy protein having 13 residues ($i = 0, 1, 2, \ldots, 12$). (a) Local feature extractor λ (with window size $w_{local} = 3$) being applied. (b) Application of the edge aggregated graph attention layer Υ to a neighborhood 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 form (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 $j \in \mathcal{N}_2$, where 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.

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 (Han and Moraga, 1995). Figure 1(c) shows the node-level classification in EGRET.

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

2.3 Overall end-to-end pipeline of EGRET

Figure 1 shows the overall end-to-end pipeline of EGRET. Here we demonstrate EGRET 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. EGRET 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 $\mathcal{N}_2 = \{1, 3, 10\}$. Note that residue 10 is not sequentially closer to node 2, but is in close proximity of node 2 in three-dimensional space.

EGRET converts the residues to a series of tokens (each token representing a residue) and uses ProtBERT (Elnaggar *et al.*, 2020) 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 feature vectors $\{\xi_{j,i} | j, i \; in \; V(G)\}$ are calculated from the structural data of the protein (available in PDB files (Berman *et al.*, 2000)). Each feature-vector $\xi_{j,i}$ is associated with one directed edge $E_{j,i}$ from node j to node i. Next, local feature extractor λ is applied to the feature-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 (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.

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

As this is the first known study on leveraging graph neural networks for PPI site prediction for single proteins, we have developed an original GAT (Veličković *et al.*, 2018) based PPI site 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 (Wang *et al.*, 2019a).

3 Results and Discussion

3.1 Dataset

We analyzed three widely used benchmark datasets, namely (1) Dset_186 (Murakami and Mizuguchi, 2010), (2) Dset_72 (Murakami and Mizuguchi, 2010), and (3) PDBset_164 (Singh *et al.*, 2014). Dset_186, Dset_72, and PDBset_164 contain 186, 72, and 164 non-repetitive protein sequences, respectively. All these three datasets have been built with proteins from PDB-database (Berman *et al.*, 2000), with sequence homology less than 25% and resolution less than 3.0 Å (solved by X-ray crystallography). We refer to Murakami and Mizuguchi (2010) and Singh *et al.* (2014) for more details. As these datasets come from different research groups, Zeng *et al.* (2020) integrated the three datasets into a fused dataset to ensure that the training set and the test set are from an identical distribution. Zeng *et al.* (2020) 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 Zeng *et al.* (2020) (available at: `https://github.com/CSUBioGroup/DeepPPISP`).

Table 1 shows the numbers of interaction and non-interaction sites in these datasets and the splits used in this study and Zeng *et al.* (2020). We note that DELPHI (Li *et al.*, 2020b), which is a sequence-based method, used a much larger training dataset containing 9,982 protein sequences. However, leveraging that large dataset for training structure-based methods is difficult due to the unavailability of curated structural information.

Table 1. Summary of the datasets analyzed in this study.

Dataset	Proteins	interaction sites	non-interaction sites
Dset_186	186	5,517 (15.23%)	30,702 (84.77%)
Dset_72	72	1,923 (10.60%)	16,217 (89.40%)
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%)

3.2 Methods

We compared our proposed EGRET and GAT-PPI (the proposed model without edge aggregation) with nine other competing methods for predicting PPI sites, namely SPPIDER (Porollo and Meller, 2007), ISIS (Ofran and Rost, 2007), PSIVER (Murakami and Mizuguchi, 2010), SPRINGS (Singh *et al.*, 2014), RF_PPI (Hou *et al.*, 2017), and especially the most recent and the most accurate predictors IntPred (Northey *et al.*, 2018), SCRIBER (Zhang and Kurgan, 2019), DeepPPISP (Zeng *et al.*, 2020) and DELPHI (Li *et al.*, 2020b). Among these nine alternate methods, SPIDER, IntPred and DeepPPISP are strucuture-based methods that leverage structural data or features derived from structural data. See supplementary materials for further details.

3.3 Evaluation metrics

For the evaluation of EGRET, we used seven widely used evaluation metrics (Li *et al.*, 2020b; Zeng *et al.*, 2020; Zhang and Kurgan, 2019), 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 supplementary materials for more details.

As rightly mentioned by Li *et al.* (2020b), AUROC and AUPRC 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 (de Vries and Bonvin, 2008; Zeng *et al.*, 2016, 2020). We performed Wilcoxon signed-rank test (Wilcoxon *et al.*, 1970) (with $\alpha = 0.05$) to measure the statistical significance of the differences between two methods.

3.4 Results on benchmark dataset

The comparison of EGRET with other state-of-the-art methods are shown in Table 2. Remarkably, EGRET outperformed all other methods under six (out of seven) evaluation metrics, including the most important ones (e.g., F1, AUROC, AUPRC, MCC). Recall is the only metric where EGRET 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, AUROC, AUPRC, MCC, which are second to only EGRET. Thus, our proposed GAT based models EGRET (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 F1-score as well as AUPRC and MCC – two of the most important evaluation metrics due to the imbalance nature of PPI prediction problem – obtained by EGRET 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). See Table S2 in supplementary materials.

Table 2. A comparison of the predictive performance of our proposed EGRET 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*
EGRET[1]	**0.715**	**0.358**	0.561	**0.438**	**0.719**	**0.405**	**0.27**

[1] Uses structural information.

[2] Results reported by DeepPPISP Zeng *et al.* (2020).

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. EGRET is substantially better than two of the most recent structure-based methods, namely DeepPPISP and IntPred. EGRET achieved 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 of the most important metrics AUPRC and MCC is quite remarkable.

In order to visually show the predicted interfacial sites, we show in Fig. 2 the true and predicted (by EGRET and DELPHI) interaction sites on a representative protein (PDB ID 3OUR, Chain-B). This protein is 150

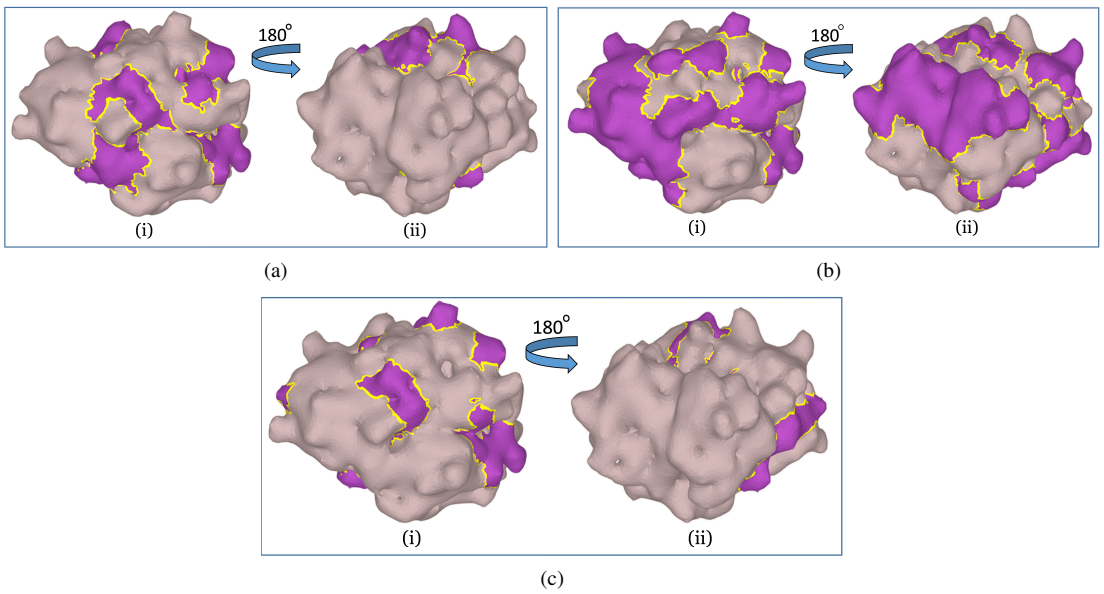

Fig. 2. Interaction sites of a representative protein in the test set (PDB ID 3OUR, Chain-B). (a) Interaction sites predicted by EGRET, (b) interaction sites predicted by DELPHI, and (c) true interactions sites as obtained from the dataset. Interaction sites are shown in purple on the protein surface. 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).

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

EGRET, unlike DeepPPISP, is designed for generating different suitable global features for different residues. Therefore, we investigated the performance of EGRET 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 interactions if they are at least 20 residues apart ($|i - j| \geq 20$), but < 8 away in terms of their atomic distances between $C\alpha$ atoms (Heffernan *et al.*, 2017). 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 EGRET, GAT-PPI and DELPHI under various model conditions in Fig. 3 (a) and Tables S3 and S4 in supplementary materials. These results show that – as expected – the performance of both EGRET and DELPHI degrades as we increase the number of non-local contacts. However, the difference in predictive performance between EGRET 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, EGRET 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.

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. 3 (b)) and Tables S5 and S6 in supplementary materials. EGRET and GAT-PPI consistently outperform DELPHI across various model conditions and the improvement tend to increase and become statistically significant as we increase the length of the proteins.

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 2. EGRET achieved better performance metrics than GAT-PPI (except for the recall). It obtained 9.5%, 12.6%, 2.1%, 0.7%, 1.8%, 7.1% performance improvement over GAT-PPI in accuracy, Precision, F1-score, 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 EGRET and GAT-PPI using both these feature sets (ProtBERT-based features and DeepPPISP features) and analyzed their predictive performance (see Table 3). The results suggest that both GAT-PPI and EGRET obtained better perfomance with the ProtBERT-based features than those achieved with DeepPPISP features.

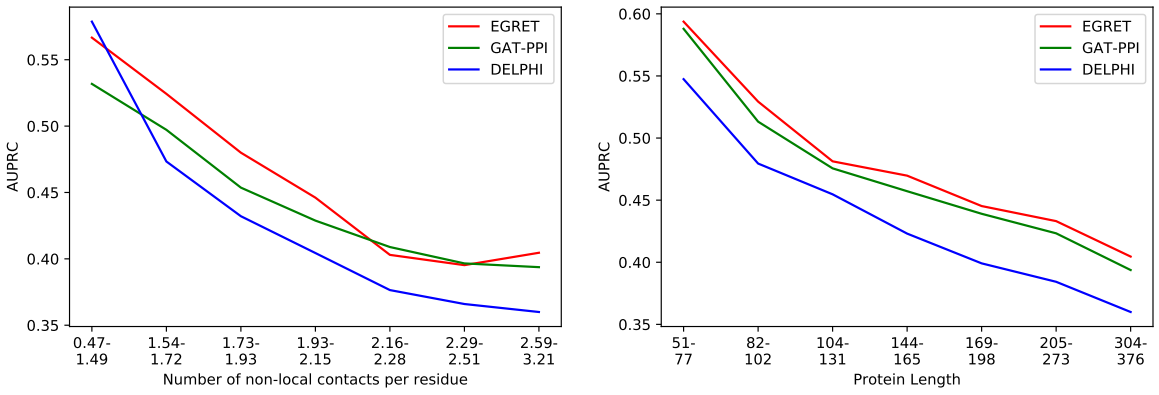

Fig. 3. Impact of long-range interactions and protein lengths on predictive performance of PPI sites prediction. (a) AUPRC of EGRET, GAT-PPI and DELPHI on varying levels of non-local interactions. (b) AUPRC of EGRET, GAT-PPI and DELPHI on varying lengths of the proteins.

Notably, EGRET consistently outperformed GAT-PPI 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), EGRET is better than or as good as DeepPPISP and DELPHI (See Tables 2 and 3).

Table 3. Impact of different types of features. We show the performance of EGRET and GAT-PPI on DeepPPISP feature set and ProtBERT-based feature set. Best results are shown in bold.

Feature-set	Method	ACC	F1	AUROC	AUPRC	MCC
DeepPPISP features	GAT-PPI	0.658	0.413	0.685	0.336	0.229
	EGRET	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
	EGRET	**0.715**	**0.438**	**0.719**	**0.405**	**0.27**

3.9 Interpretability

We investigated the interpretability of EGRET to provide insights on how the architecture is making decisions. Some recent studies demonstrated partial interpretability of deep neural networks for solving various problems in computational biology (in particular, see Uddin *et al.* (2020) and Vig *et al.* (2020)).

3.9.1 Interpretability of edges

Let the EGRET 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 test set, 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 between 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 EGRET, if there is an edge between them.

3.9.2 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 EGRET – 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. 4(a)).

Figure 4(a) shows the neighborhood $\mathcal{N}_{90}$ of residue 90, and Fig. 4(b) shows the weights (attention scores) of the edges in Fig. 4(b) using color gradient. Deeper hues indicate higher levels of attention. This residue is an interaction site and EGRET rightly predicted this. The other interaction sites, predicted by EGRET, in this neighborhood are shown in green. Interestingly, the edges with relatively higher attention scores (darker color hues) are mostly associated with source 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 EGRET does not solely depend on the attention scores as

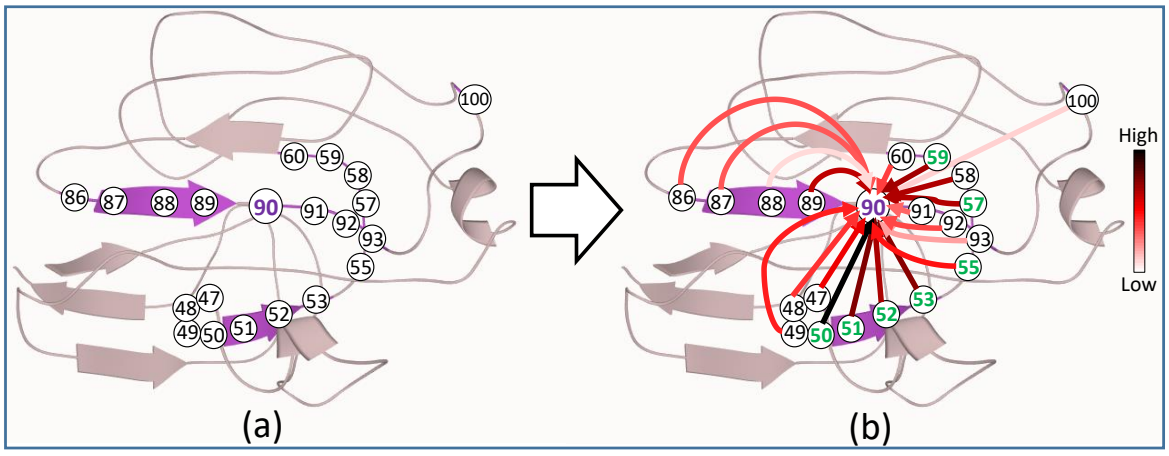

Fig. 4. Interpretability of the edge features produced by EGRET in predicting the interaction sites. (a) A cartoon images of a protein (PDB ID 3OUR, Chain B), where we show the neighborhood (window size = 20) of the residue 90. (b) The attention scores along the edges in this neighborhood using a color 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 EGRET.

it rightly predicted residue 58 and 89 to be non-interaction sites despite their associate edges having 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.

3.10 Running time

We investigated the time required to generate the features used by EGRET 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).

EGRET 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 4). 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 more time-consuming. Moreover, PPI prediction time of EGRET, given the generated features, is also faster than DeepPPISP.

Table 4. Running time comparison between EGRET and DeepPPISP. We show the times (in seconds) 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	EGRET	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	EGRET	4.57±0.05	11.84±0.02	16.41±0.07
	DeepPPISP	12.21±0.15	3706.74±6	3718.95±6.15

4 Conclusions

We have presented EGRET, 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. The demonstrated performance improvement across all seven evaluation metrics and the partial interpretability of EGRET are quite promising. Thus, we believe EGRET advances the state-of-the-art in this domain, and will be considered as a useful tool for PPISP.

References

Ahmad, S. and Mizuguchi, K. (2011). Partner-aware prediction of interacting residues in protein-protein complexes from sequence data. *PLoS One*, **6**(12), e29104.

Ahmed, K. S., Saloma, N. H., and Kadah, Y. M. (2011). Improving the prediction of yeast protein function using weighted protein-protein interactions. *Theoretical Biology and Medical Modelling*, **8**(1), 11.

Altschul, S. F., Madden, T. L., Schäffer, A. A., Zhang, J., Zhang, Z., Miller, W., and Lipman, D. J. (1997). Gapped blast and psi-blast: a new generation of protein database search programs. *Nucleic Acids Research*, **25**(17), 3389–3402.

Aumentado-Armstrong, T. T., Istrate, B., and Murgita, R. A. (2015). Algorithmic approaches to protein-protein interaction site prediction. *Algorithms for Molecular Biology*, **10**(1), 7.

Bahdanau, D., Cho, K., and Bengio, Y. (2015). Neural machine translation by jointly learning to align and translate. In *3rd International Conference on Learning Representations, ICLR 2015*.

Berman, H. M., Westbrook, J., Feng, Z., Gilliland, G., Bhat, T. N., Weissig, H., Shindyalov, I. N., and Bourne, P. E. (2000). The protein data bank. *Nucleic Acids Research*, **28**(1), 235–242.

Brettner, L. M. and Masel, J. (2012). Protein stickiness, rather than number of functional protein-protein interactions, predicts expression noise and plasticity in yeast. *BMC Systems Biology*, **6**(1), 128.

Chen, H. and Zhou, H.-X. (2005). Prediction of interface residues in protein–protein complexes by a consensus neural network method: test against nmr data. *Proteins: Structure, Function, and Bioinformatics*, **61**(1), 21–35.

Dai, Z., Yang, Z., Yang, Y., Carbonell, J. G., Le, Q., and Salakhutdinov, R. (2019). 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.

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

de Vries, S. J. and Bonvin, A. M. (2008). How proteins get in touch: interface prediction in the study of biomolecular complexes. *Current Protein and Peptide Science*, **9**(4), 394–406.

picture(0,0)(-35,0)(1,0)30 (0,35)(0,-1)30 picture

picture(0,0)(35,0)(-1,0)30 (0,35)(0,-1)30 picture

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2019). 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. Association for Computational Linguistics.

Elnaggar, A., Heinzinger, M., Dallago, C., Rihawi, G., Wang, Y., Jones, L., Gibbs, T., Feher, T., Angerer, C., Bhowmik, D., *et al.* (2020). Prottrans: Towards cracking the language of life's code through self-supervised deep learning and high performance computing. *arXiv preprint arXiv:2007.06225*.

Eppstein, D., Paterson, M. S., and Yao, F. F. (1997). On nearest-neighbor graphs. *Discrete & Computational Geometry*, **17**(3), 263–282.

Ezkurdia, I., Bartoli, L., Fariselli, P., Casadio, R., Valencia, A., and Tress, M. L. (2009). Progress and challenges in predicting protein–protein interaction sites. *Briefings in Bioinformatics*, **10**(3), 233–246.

Fernandez-Recio, J., Totrov, M., and Abagyan, R. (2004). Identification of protein–protein interaction sites from docking energy landscapes. *Journal of molecular biology*, **335**(3), 843–865.

Fout, A., Byrd, J., Shariat, B., and Ben-Hur, A. (2017). Protein interface prediction using graph convolutional networks. In *Advances in neural information processing systems*, pages 6530–6539.

Giot, L., Bader, J. S., Brouwer, C., Chaudhuri, A., Kuang, B., Li, Y., Hao, Y., Ooi, C., Godwin, B., Vitols, E., *et al.* (2003). A protein interaction map of drosophila melanogaster. *Science*, **302**(5651), 1727–1736.

Hamp, T. and Rost, B. (2015). More challenges for machine-learning protein interactions. *Bioinformatics*, **31**(10), 1521–1525.

Han, J. and Moraga, C. (1995). The influence of the sigmoid function parameters on the speed of backpropagation learning. In *International Workshop on Artificial Neural Networks*, pages 195–201. Springer.

Hanson, J., Paliwal, K., Litfin, T., Yang, Y., and Zhou, Y. (2019). 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.

Heffernan, R., Yang, Y., Paliwal, K., and Zhou, Y. (2017). 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.

Hou, Q., De Geest, P. F., Vranken, W. F., Heringa, J., and Feenstra, K. A. (2017). Seeing the trees through the forest: sequence-based homo-and heteromeric protein-protein interaction sites prediction using random forest. *Bioinformatics*, **33**(10), 1479–1487.

Jones, S. and Thornton, J. M. (1997). Analysis of protein-protein interaction sites using surface patches. *Journal of molecular biology*, **272**(1), 121–132.

Kipf, T. N. and Welling, M. (2017). Semi-supervised classification with graph convolutional networks. In *International Conference on Learning Representations (ICLR)*.

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

La, D. and Kihara, D. (2012). A novel method for protein–protein interaction site prediction using phylogenetic substitution models. *Proteins: Structure, Function, and Bioinformatics*, **80**(1), 126–141.

Lan, Z., Chen, M., Goodman, S., Gimpel, K., Sharma, P., and Soricut, R. (2019). Albert: A lite bert for self-supervised learning of language representations. In *International Conference on Learning Representations*.

Li, X., Li, W., Zeng, M., Zheng, R., and Li, M. (2020a). Network-based methods for predicting essential genes or proteins: a survey. *Briefings in bioinformatics*, **21**(2), 566–583.

Li, Y., Golding, G. B., and Ilie, L. (2020b). DELPHI: accurate deep ensemble model for protein interaction sites prediction. *Bioinformatics*. btaa750.

Liu, Y., Yuan, H., Cai, L., and Ji, S. (2020). 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.

Mihel, J., Šikić, M., Tomić, S., Jeren, B., and Vlahoviček, K. (2008). Psaia–protein structure and interaction analyzer. *BMC Structural Biology*, **8**(1), 21.

Murakami, Y. and Mizuguchi, K. (2010). Applying the naïve bayes classifier with kernel density estimation to the prediction of protein–protein interaction sites. *Bioinformatics*, **26**(15), 1841–1848.

Nibbe, R. K., Chowdhury, S. A., Koyutürk, M., Ewing, R., and Chance, M. R. (2011). Protein–protein interaction networks and subnetworks in the biology of disease. *Wiley Interdisciplinary Reviews: Systems Biology and Medicine*, **3**(3), 357–367.

Northey, T. C., Barešić, A., and Martin, A. C. (2018). Intpred: a structure-based predictor of protein–protein interaction sites. *Bioinformatics*, **34**(2), 223–229.

Ofran, Y. and Rost, B. (2007). Isis: interaction sites identified from sequence. *Bioinformatics*, **23**(2), e13–e16.

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

Petta, I., Lievens, S., Libert, C., Tavernier, J., and De Bosscher, K. (2016). Modulation of protein–protein interactions for the development of novel therapeutics. *Molecular Therapy*, **24**(4), 707–718.

Porollo, A. and Meller, J. (2007). Prediction-based fingerprints of protein–protein interactions. *Proteins: Structure, Function, and Bioinformatics*, **66**(3), 630–645.

Rules, I.-I. T. (1969). A one letter notation for amino acid sequence. *Biochem. J*, **113**, 1–4.

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

Sperandio, O. (2012). Toward the design of drugs on protein-protein interactions. *Current Pharmaceutical Design*, **18**(30), 4585.

Terentiev, A., Moldogazieva, N., and Shaitan, K. (2009). Dynamic proteomics in modeling of the living cell. protein-protein interactions. *Biochemistry (Moscow)*, **74**(13), 1586–1607.

Townshend, R., Bedi, R., Suriana, P., and Dror, R. (2019). End-to-end learning on 3d protein structure for interface prediction. In *Advances in Neural Information Processing Systems*, pages 15642–15651.

Uddin, M. R., Mahbub, S., Rahman, M. S., and Bayzid, M. S. (2020). SAINT: self-attention augmented inception-inside-inception network improves protein secondary structure prediction. *Bioinformatics*, **36**(17), 4599–4608.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I. (2017). Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008.

Veličković, P., Cucurull, G., Casanova, A., Romero, A., Liò, P., and Bengio, Y. (2018). Graph attention networks. In *International Conference on Learning Representations*.

Vig, J., Madani, A., Varshney, L. R., Xiong, C., Socher, R., and Rajani, N. F. (2020). Bertology meets biology: Interpreting attention in protein language models. *arXiv preprint arXiv:2006.15222*.

Wang, M., Yu, L., Zheng, D., Gan, Q., Gai, Y., Ye, Z., Li, M., Zhou, J., Huang, Q., Ma, C., *et al.* (2019a). Deep graph library: Towards efficient and scalable deep learning on graphs. *arXiv preprint arXiv:1909.01315*.

Wang, X., Yu, B., Ma, A., Chen, C., Liu, B., and Ma, Q. (2019b). Protein–protein interaction sites prediction by ensemble random forests with synthetic minority oversampling technique. *Bioinformatics*, **35**(14), 2395–2402.

Wang, X., Yu, B., Ma, A., Chen, C., Liu, B., and Ma, Q. (2019c). Protein–protein interaction sites prediction by ensemble random forests with synthetic minority oversampling technique. *Bioinformatics*, **35**(14), 2395–2402.

Wang, Y., Sun, Y., Liu, Z., Sarma, S. E., Bronstein, M. M., and Solomon, J. M. (2019d). Dynamic graph cnn for learning on point clouds. *Acm Transactions On Graphics (tog)*, **38**(5), 1–12.

Wei, Z.-S., Han, K., Yang, J.-Y., Shen, H.-B., and Yu, D.-J. (2016). Protein–protein interaction sites prediction by ensembling svm and sample-weighted random forests. *Neurocomputing*, **193**, 201–212.

Wilcoxon, F., Katti, S., and Wilcox, R. A. (1970). 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.

Wodak, S. J., Vlasblom, J., Turinsky, A. L., and Pu, S. (2013). Protein–protein interaction networks: the puzzling riches. *Current Opinion in Structural Biology*, **23**(6), 941–953.

Yan, C., Dobbs, D., and Honavar, V. (2004). A two-stage classifier for identification of protein–protein interface residues. *Bioinformatics*, **20**(suppl_1), i371–i378.

Yang, Z., Dai, Z., Yang, Y., Carbonell, J., Salakhutdinov, R. R., and Le, Q. V. (2019). Xlnet: Generalized autoregressive pretraining for language understanding. In *Advances in neural information processing systems*, pages 5753–5763.

Zeng, M., Zou, B., Wei, F., Liu, X., and Wang, L. (2016). 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.

Zeng, M., Zhang, F., Wu, F.-X., Li, Y., Wang, J., and Li, M. (2020). Protein–protein interaction site prediction through combining local and global features with deep neural networks. *Bioinformatics*, **36**(4), 1114–1120.

Zhang, B., Li, J., Quan, L., Chen, Y., and Lü, Q. (2019). Sequence-based prediction of protein-protein interaction sites by simplified long short-term memory network. *Neurocomputing*, **357**, 86–100.

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

Zitnik, M. and Leskovec, J. (2017). Predicting multicellular function through multi-layer tissue networks. *Bioinformatics*, **33**(14), i190–i198.

picture(0,0)(-35,0)(1,0)30 (0,-35)(0,1)30 picture

picture(0,0)(35,0)(-1,0)30 (0,-35)(0,1)30 picture