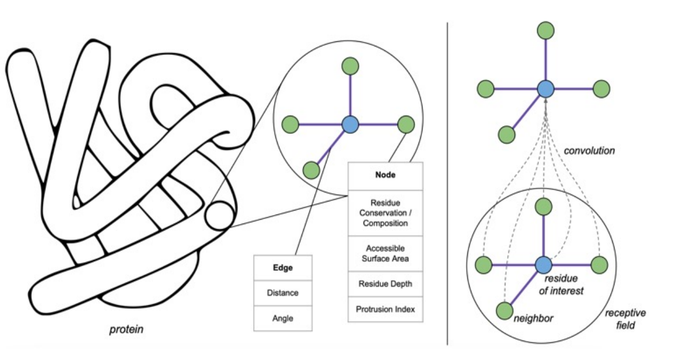
PROTEIN INTERFACE PREDICTION: AN ANALYSIS OF LIMITATIONS FOR EXISTING MODELS

Lige Zhang, Ziyu Huang
STATS 402, 2024 Spring S2

Introduction

Prediction of protein interfaces is fundamentally important for comprehending biological processes and advancing biotechnological applications. Prior studies highlight the complexity and importance of accurately identifying protein-protein interactions, which is essential for drug discovery, indicating a need for precise and sturdy protein interface prediction (PIP) models. However, the utility of PIP is as a preliminary tool, guiding further empirical validation, such as NMR to test interfaces in large protein complexes, and SPR for studying interactions with minimal material requirements, despite significant associated costs of these methods. Advancements in PIP require improved model accuracy due to the high expense of experimental testing.

With recent applications of deep learning in scientific inquiry, despite issues with interpretability, accuracy in these models has led to their increased use in this domain. This research introduces a graph neural network (GNN)-based deep learning approach for PIP, chosen for the rich structural data in proteins and GNN's ability to mine topological information. We've split the PIP challenge into two tasks: extracting topological features of ligand/receptor proteins using GNN, and employing a multilayer perceptron (MLP) for binary classification to determine interface sites.



Materials and Methods

To predict a potential protein interface, the model needs a pair of two proteins. It takes in two protein graphs as input, receptor graph, and ligand graph. Then GNN modules parallelly extract topological features for each of them. The next step involves building nodes pairs. This process pairs a node from the receptor graph with a node from the ligand graph. This systematic pairing generates all possible combinations of node pairs across the two graphs. Finally, the MLP module solves a binary classification task. For example, it takes a vector CONCAT(R_m,L_n), where (R_m,L_n) represents a node pair, and performs a binary classification to predict whether this pair is an interface locus or not.

This is the principle for GCN module:

$$\begin{cases} m_i^{t+1} = W^t x_i^t + b^t \ \ (Feature\ Transform) \\ x_i^{t+1} = \sigma\left(m_i^{t+1} + \sum_{j \in N(i)} w_{ij}\, m_j^{t+1} \right) \ \ (Neighborhood\ Aggregation) \end{cases}$$

This is the updating equation for s1-GCN layer:

$$h_v^{(k+1)} = \sigma\left(W_{self}\, h_v^{(k)} + \sum_{u \in N(v)} W_{neigh} h_u^{(K)} + \mathbf{1} \cdot \sum_{u \in N(v)} w_{edge}^T e_u \right)$$

This is the updating equation for s2-GCN layer:

$$h_v^{(k+1)} = \sigma\left(W_{self}\, h_v^{(k)} + \frac{1}{|N(v)|}\left(\sum_{u \in N(v)} W_{neigh} h_u^{(K)} + \mathbf{1} \cdot \sum_{u \in N(v)} w_{edge}^T e_u \right) \right)$$

This is the updating equation for s1-GAT layer:

$$h_v^{(k+1)} = \sigma\left(W_{self}\, h_v^{(k)} + \sum_{u \in N(v)} \alpha_{vu} W_{neigh} h_u^{(K)} + \frac{1}{|N(v)|}\left(\mathbf{1} \cdot \sum_{u \in N(v)} w_{edge}^T e_u \right) \right)$$

This is the updating equation for s2-GAT layer:

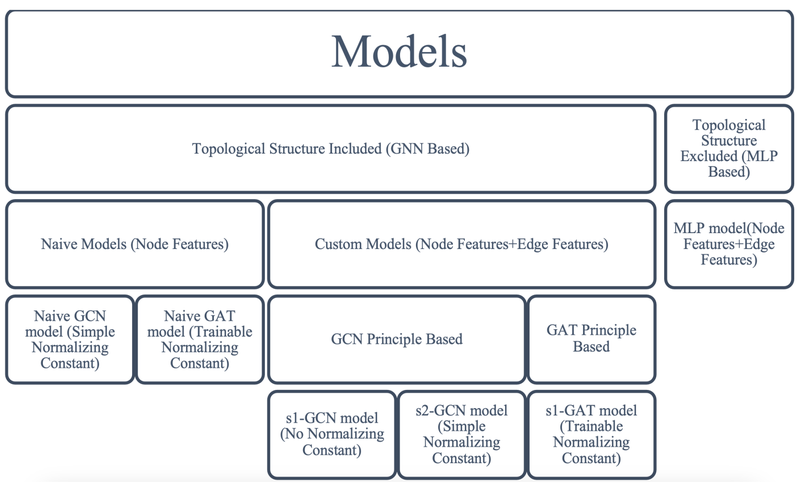
$$h_v^{(k+1)} = \sigma\left(\frac{1}{I} \sum_{i=1}^{I} \left(W_{self}\, h_v^{(k)} + \sum_{u \in N(v)} \alpha_{vu} W_{neigh} h_u^{(K)} + \frac{1}{|N(v)|}\left(\mathbf{1} \cdot \sum_{u \in N(v)} w_{edge}^T e_u \right) \right) \right)$$

This is the how attention is calculated:

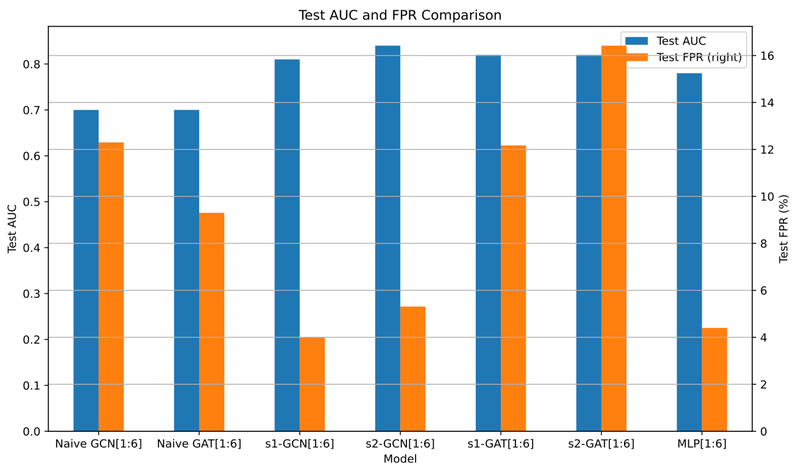
$$\alpha_{vu} = softmax_{u \in N(v)}\left(LeakyReLU\left(W_{attention} W_{neigh} h_u \right) \right)$$

This is the weighted cross-entropy loss function:

$$WCE(p, \hat{p}) = -(\beta p log(\hat{p}) + (1 - p) log(1 - \hat{p}))$$



Results



When merely using node features as predictors for the model, like the naïve GCN and the naïve GAT model, the generalizations are not desirable. When combining node features and edge features, like the remaining models, they provide obvious improved performance. A general trend to recognize is that the increment of AUC is often accompanied with a decrement in FPR. However, it is also observed that AUC and FPR do not have a monotonic relationship. Although s-GAT models are more complex and flexible than s-GCN models, they provide a worse performance instead. Although MLP model has the simplest architecture and completely ignores the topological features of a protein, its performance is compatible with others.

Weight [Pos:Neg]	*AUC*	*FPR*	*# of TP*
S2-GCN Model [1:6]	0.84	5.308%	1813
[1:2]	0.82	4.928%	1651
[1:1]	0.83	0.637%	447
[3:1]	0.83	0.035%	64
[5:1]	0.83	0.007%	9
S2-GAT Model [1:6]	0.82	16.420%	3082
[1:1]	0.83	0.725%	514
[3:1]	0.83	0.000%	5

Table n: Experiments on Different Weights:
# of TP means the number of true positive predictions

Hid Dim\Criterion	*AUC*	*FPR*	*# of TP*
20	0.83	0.007%	7
32	0.83	0.035%	64
50	0.81	3.252%	224
60	0.81	0.000%	0
75	0.80	0.003%	6

Table n: Hidden Dimension

It can be shown that choosing different weights in the loss function rarely influences the ultimate test AUC but have a large impact on the FPR. As the penalty for misclassifying a positive sample increase, FPR drops quickly. However, the model tends to predict more samples as negative cases as well. There is a trade-off between one wants the model to have an accurate identification and one wants the model to reduce false positive predictions. In terms of hidden dimension, the results are similar. Regardless of the change of hidden dimension, the AUC on the test set always remains to be approximately the same, and the FPR varies significantly.

Weight [Pos:Neg]	*Model*	*AUR*	*FPR*
[1:6]	S2-GCN	0.84	5.31%
	MLP	0.78	4.40%
[1:1]	S2-GCN	0.83	0.64%
	MLP	0.80	3.60%
[5:1]	S2-GCN	0.83	0.01%
	MLP	0.81	0.71%

Table n: A Comparison between s2-GCN model and MLP model

It can be shown by table n that MLP model has little difference from s2-GCN model. This result indicates that current graph representation is not effective enough to capture the real topological structure of a protein.

Discussion

In current models, all graphs are embedded in the Euclidean space. However, we could potentially embed protein graphs into spherical space because some proteins, like virus protein shells, exhibit to be a sphere-like structure, or hyperbolic space because the underlying graph for protein interaction exabits hierarchical structure. However, many protein structures remain to be unclear at current stage, so it is not clear which embedding space is suitable for PIP task. Additionally, most existing operations are not properly defined in Non-Euclidean spaces.

It is also possible to use products like AlphaFold to simulate protein structures, to obtain potential training data. The underlying problem for this approach could be that the training data can be inaccurate because AlphaFold does not guarantee 100% correct structure prediction. However, this could still be a potential working direction.

Conclusion

In this research, we have introduced s-GCN/s-GAT a novel graph convolution layer to improve feature extraction in protein interface prediction task. The performed model desirable performance, and we have in-depth analyzed how hyperparameters influence the model to give false positive predictions, which shed light on future research. By a comparison with MLP model, we further show the limitation of current models, that current graph construction or embedding method is not a desirable representation of a real protein structure. This could be an important future working direction because it lays the foundation for other downstream tasks.

昆山杜克大学
DUKE KUNSHAN UNIVERSITY

Visualization