

Protein Interface Prediction using Graph Convolutional Networks

NeurIPS 2019

Presenter : Eli Draizen

University of Virginia

<https://qdata.github.io/deep2Read>

Outline

1 Introduction

2 Graph Convolutional Neural Networks

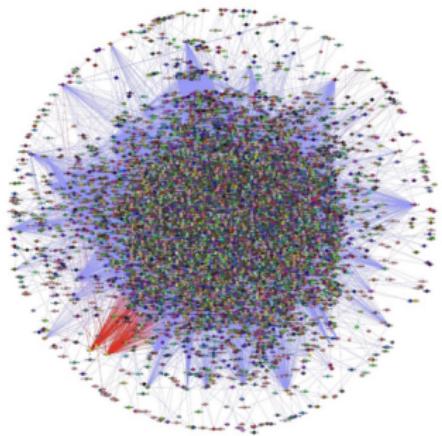
- Node Averaging
- Node and Edge Averaging
- Order Dependent

3 Interface Predictor

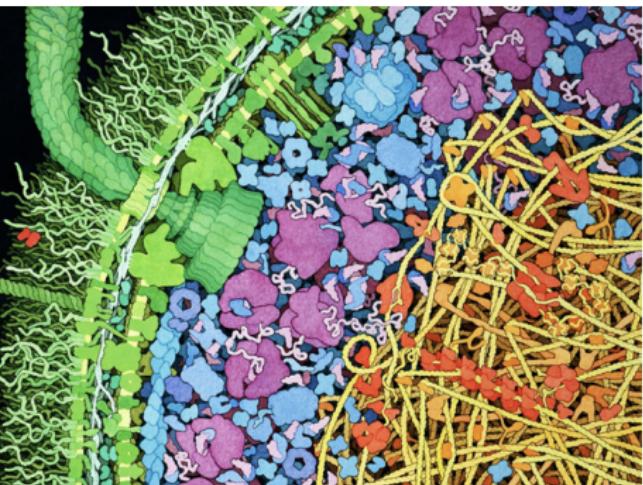
- Model
- Results

Protein-Protein Interactions

Protein-protein interactions (PPI) mediate many biological functions and are crucial to understanding biological pathways.



(a) PPI Network



(b) Molecular Crowding by David Goodsell

Goal of the paper

Given two proteins (not in a complex), can you predict the amino acids that interact with each other?

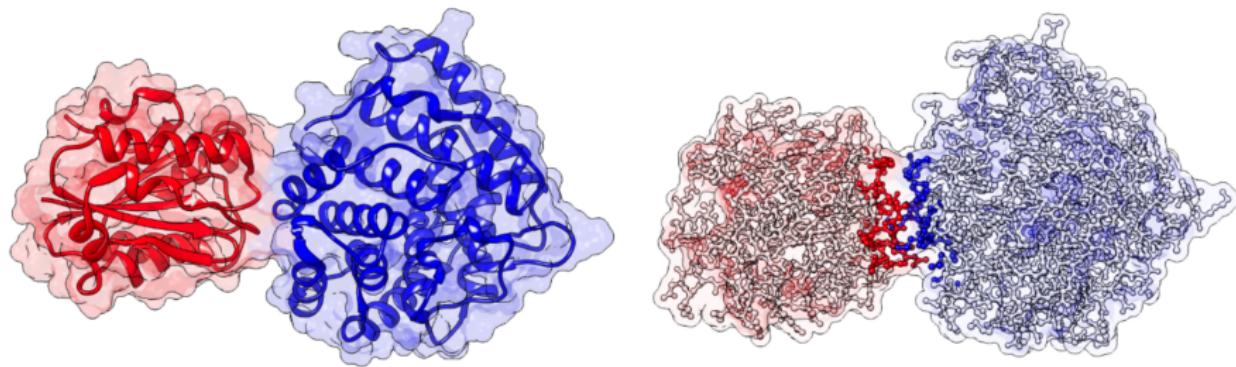
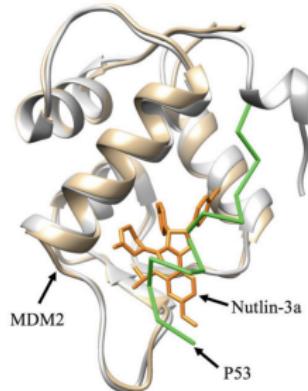


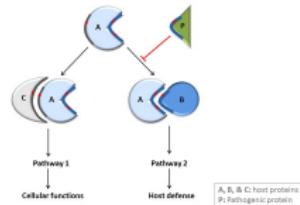
Figure: PDB ID 4M76. Modified from Fout et al. NeurIPS 2017 poster.

Importance



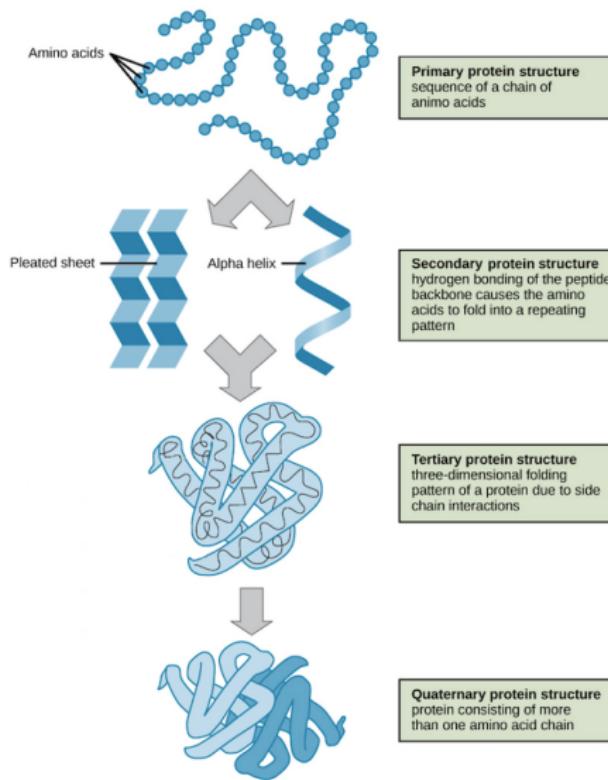
(a) Drug development - inhibit PPIs.
Goncaerenco 2017

Interface mimicry allows pathogenic proteins to rewrite the host networks

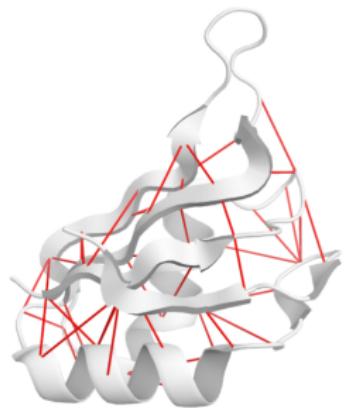


(b) Identify Pathogenic Proteins that mimic human interactions. Nussinov 2017

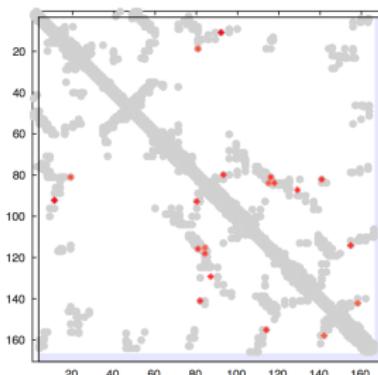
Protein Structure Primer



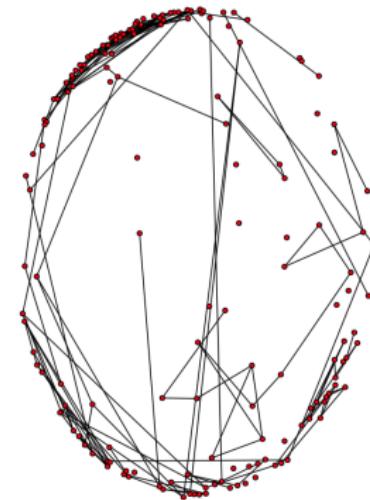
Protein Structure Representations



(a) Full 3D Structure



(b) Contact Map



(c) Graph

Figure: Figures modified from John Ingraham and Debbie Marks

Protein Graph Features

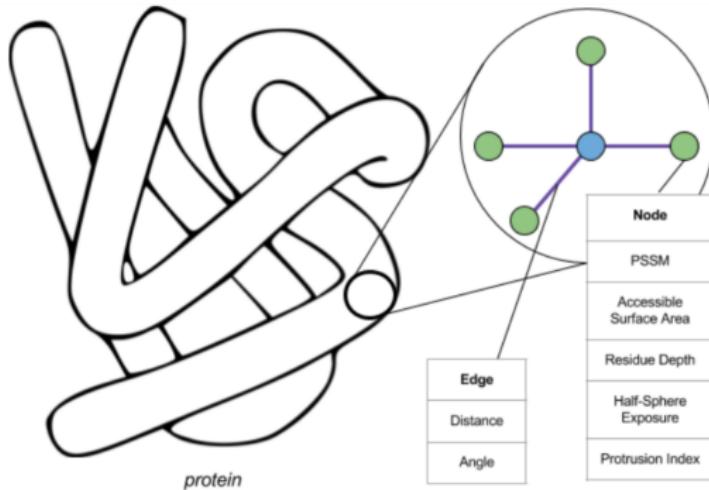


Figure: Proteins represented as fully connected graph. Edge features contain the distance as a Radial Basis Function (RBF) with std. dev. of 18 Angstroms

Outline

1 Introduction

2 Graph Convolutional Neural Networks

- Node Averaging
- Node and Edge Averaging
- Order Dependent

3 Interface Predictor

- Model
- Results

Graph Convolutions

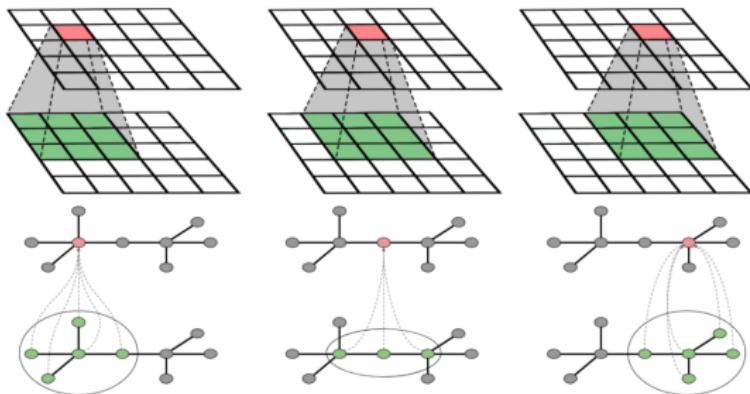


Figure: Convolution on a grid vs a graph. Receptor field (number of adjacent nodes to the center node; closest in 3D space) is a hyper parameter with 21 being best. Modified from Fout et al. NeurIPS 2017 poster.

Outline

1 Introduction

2 Graph Convolutional Neural Networks

- Node Averaging
- Node and Edge Averaging
- Order Dependent

3 Interface Predictor

- Model
- Results

Node Averaging

$$z_i = \sigma \left(W^C x_i + \frac{1}{|\mathcal{N}_i|} \sum_{j \in \mathcal{N}_i} W^N x_j + b \right) \quad (1)$$

\mathcal{N}_i is the set of neighbors of node i

W^C is the weight matrix associated with the center node,

W^N is the weight matrix associated with neighboring nodes,

b is a vector of biases, one for each filter.

Outline

1 Introduction

2 Graph Convolutional Neural Networks

- Node Averaging
- Node and Edge Averaging
- Order Dependent

3 Interface Predictor

- Model
- Results

Node and Edge Averaging (Order Independent)

$$z_i = \sigma \left(W^C x_i + \frac{1}{|\mathcal{N}_i|} \sum_{j \in \mathcal{N}_i} W^N x_j + \frac{1}{|\mathcal{N}_i|} \sum_{j \in \mathcal{N}_i} W^E A_{ij} + b \right) \quad (2)$$

\mathcal{N}_i is the set of neighbors of node i

W^C is the weight matrix associated with the center node,

W^N is the weight matrix associated with neighboring nodes,

b is a vector of biases, one for each filter,

W^E is the weight matrix associated with edge features.

A_{ij} the feature vector associated with the edge between nodes i and j

Outline

1 Introduction

2 Graph Convolutional Neural Networks

- Node Averaging
- Node and Edge Averaging
- Order Dependent

3 Interface Predictor

- Model
- Results

Order Dependent

$$z_i = \sigma \left(W^C x_i + \frac{1}{|\mathcal{N}_i|} \sum_{j \in \mathcal{N}_i} W_j^N x_j + \frac{1}{|\mathcal{N}_i|} \sum_{j \in \mathcal{N}_i} W_j^E A_{ij} + b \right) \quad (3)$$

\mathcal{N}_i is the set of neighbors of node i,

W_j^C is the weight matrix associated with the jth node or the edges connecting to the jth node,

W_j^N is the weight matrix associated with the jth node or the edges connecting to the jth node,

b is a vector of biases, one for each filter.

W^E is the weight matrix associated with edge features.

A_{ij} the feature vector associated with the edge between nodes i and j

Outline

1 Introduction

2 Graph Convolutional Neural Networks

- Node Averaging
- Node and Edge Averaging
- Order Dependent

3 Interface Predictor

- Model
- Results

Outline

1 Introduction

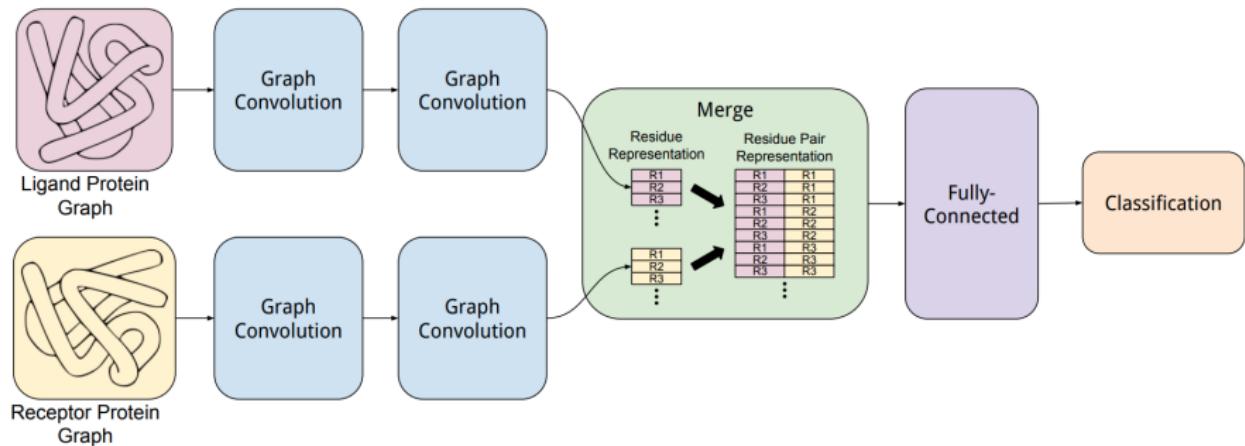
2 Graph Convolutional Neural Networks

- Node Averaging
- Node and Edge Averaging
- Order Dependent

3 Interface Predictor

- Model
- Results

Interface Predictor - GCN Model



Data Partition	Complexes	Positive examples	Negative examples
Train	140	12,866 (9.1%)	128,660 (90.9%)
Validation	35	3,138 (0.2%)	1,874,322 (99.8%)
Test	55	4,871 (0.1%)	4,953,446 (99.9%)

Outline

1 Introduction

2 Graph Convolutional Neural Networks

- Node Averaging
- Node and Edge Averaging
- Order Dependent

3 Interface Predictor

- Model
- Results

Results

No Convolution
$$z_i = \sigma\left(Wx_i + b\right)$$

Single Weight Matrix [2]
$$z_i = \sigma\left(Wx_i + \frac{1}{|\mathcal{N}_i|} \sum_{j \in \mathcal{N}_i} Wx_j + b\right)$$

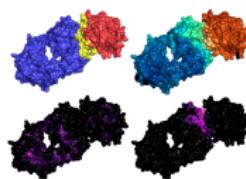
Node Average
$$z_i = \sigma\left(W^C x_i + \frac{1}{|\mathcal{N}_i|} \sum_{j \in \mathcal{N}_i} W^N x_j + b\right)$$

Node and Edge Average
$$z_i = \sigma\left(W^C x_i + \frac{1}{|\mathcal{N}_i|} \sum_{j \in \mathcal{N}_i} W^N x_j + \frac{1}{|\mathcal{N}_i|} \sum_{j \in \mathcal{N}_i} W^E A_{ij} + b\right)$$

Order Dependent
$$z_i = \sigma\left(W^C x_i + \frac{1}{|\mathcal{N}_i|} \sum_{j \in \mathcal{N}_i} W_j^N x_j + \frac{1}{|\mathcal{N}_i|} \sum_{j \in \mathcal{N}_i} W_j^E A_{ij} + b\right)$$

Deep Tensor Networks
(DTNN) [4]
$$z_i = x_i + \frac{1}{|\mathcal{N}_i|} \sum_{j \in \mathcal{N}_i} \sigma\left(W \left[(W^N x_j + b^N) \odot (W^E A_{ij} + b^E) \right]\right)$$

Method	Convolutional Layers			
	1	2	3	4
No Convolution	0.812 (0.007)	0.810 (0.006)	0.808 (0.006)	0.796 (0.006)
Diffusion (DCNN) (2 hops) [1]	0.790 (0.014)	—	—	—
Diffusion (DCNN) (5 hops) [1])	0.828 (0.018)	—	—	—
Single Weight Matrix	0.865 (0.007)	0.871 (0.013)	0.873 (0.017)	0.869 (0.017)
Node Average	0.864 (0.007)	0.882 (0.007)	0.891 (0.005)	0.889 (0.005)
Node and Edge Average	0.876 (0.005)	0.898 (0.005)	0.895 (0.006)	0.889 (0.007)
Order Dependent	0.854 (0.004)	0.873 (0.005)	0.891 (0.004)	0.889 (0.008)
DTNN [4]	0.867 (0.007)	0.880 (0.007)	0.882 (0.008)	0.873 (0.012)



Conclusion