Quality prediction of proteins models with spherical convolutions on three-dimensional graphs

Nikita Pavlichenko

Moscow Institute of Physics and Technology

Course: My first scientific paper
Group 793, 2020
Consultants: I. Igashov, S. Grudinin

Goal of research

Goals

- Develop the new type of convolution operations on three-dimensional graphs
- Apply this method to Protein Quality Assessment problem

Problem statement

Problem statement

- Protein can be represented as a graph - a chain of amino acids
- Its properties are determined by its folding
- We have lots of folding models and want to choise the best
- We need to predict quality of these models - regression problem on graph
- MSE seems the most suitable loss function

Solution

Graph Convolutional Networks

- A very common method is Graph Convolutional Network
- l-th layer can be represented as $H^{(l)} = AH^{(l-1)}W^{(l)}$, where A is the adjacency matrix, $W^{(l)}$ is weights matrix

Props and Cons

- It was successfully applied for PQA before
- It does not capture a local protein structure

Spherical convolutions

- Amino acids are connected one by another so we can define a local coordinate system for every amino acid
- So, let's project it's neighbors onto a unit sphere.
- Conider a function of spherical coordinates. It can be expaned as a series of spherical harmonics.
- $f(\phi, \psi) = \sum_{l=0}^{\infty} \sum_{m=-l}^{m=l} f_l^m Y_l^m(\phi, \psi)$
- Leave only a few coeficients and write it in a matrix view
- $f(\Omega) \approx f_W(\Omega) = \sum_{l=0}^{L} \sum_m \boldsymbol{W}_l^m Y_l^m(\Omega)$
- Now we can introduce Spherical Convolution operation
- $f_W \circ v_i = \sum_{v_j \in \mathcal{N}(v_i)} f_W(\Omega_i^j) x(v_i)$

Solution

Spherical Convolutional Network

- Spherical convolution layer:
  $$\boldsymbol{X} \longrightarrow \boldsymbol{X}' = \sigma(f_W \circ \boldsymbol{X}) = \sigma\left(\sum_{l,m} Y_l^m(\boldsymbol{A}_\Omega)\boldsymbol{X}\boldsymbol{W}_l^m\right)$$
- Learn $\boldsymbol{W}_l^m$ matrices using Adam optimizer

Computational experiment

Dataset

- We use CASP competition data. CASP8-CASP11 as a train sample and CASP12 as a testing sample.
- Spherical harmonics are precalculated: we use an order of 5 and it takes 60GB of HDD

Baseline

- 8 GCN layers with 4 linear layers in the beggining as an encoder and 5 linear layers in the end is the best setup for GCN
- Learn on CPU since the loading from HDD is a bottleneck

Computational experiment

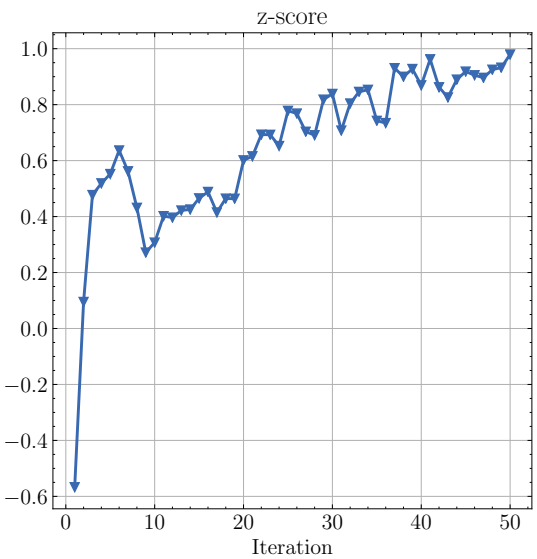
Algorithm

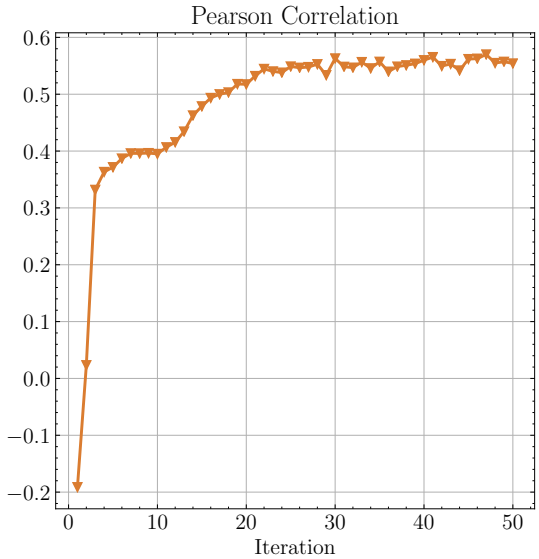
- Optimal architecture is 5 spherical convolution layers
- All code is written in PyTorch
- Learn on GPU in 4 parallel subprocesses

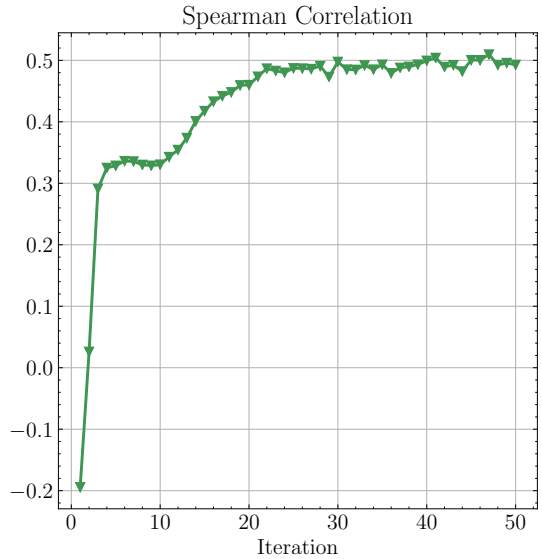
Results

- The common choice for a quality metric is Pearson correlation between ground truth and predictions
- The best achieved result for GCN is 0.57 comaring with 0.77 with spherical convolutions

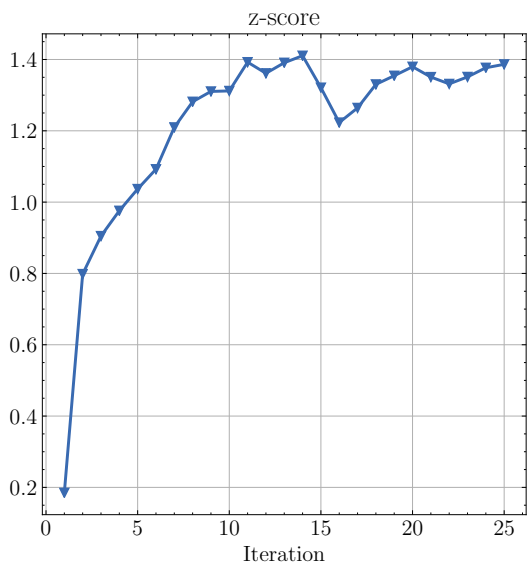
Computational experiment

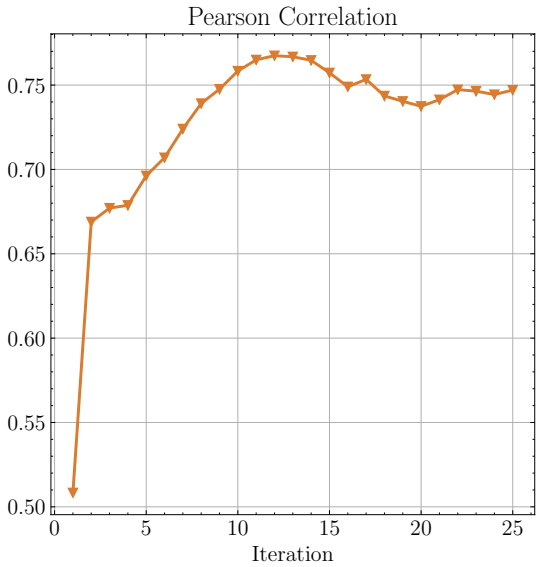


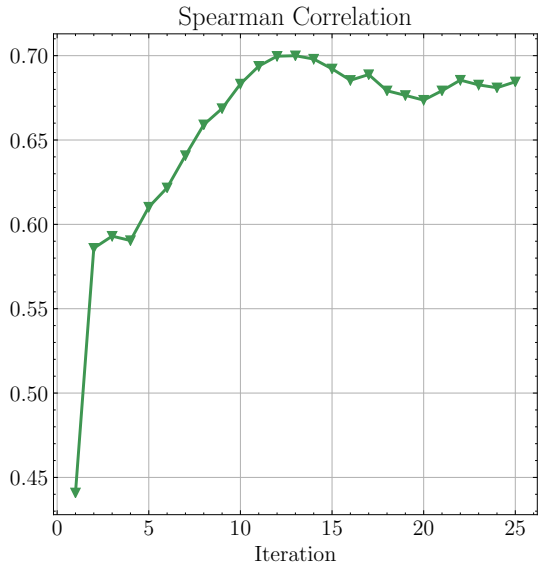




Computational experiment







Conclusion

- For the first time we introduced spherical convolution operation
- It has shown a significant improvement in Protein Quality Assessment
- It potentially has a huge space for improvements e.g. variational autoencoders