SBI/Python Project: Technical Summary of SiteScanner

Romain Pastre, Leyao Jin, Daniel Guiñón

April 22, 2025

Introduction

In the present work, we propose **SiteScanner**, a deep learning-based framework to predict binding region in protein-ligand complexes. More precisely, we build a graph neural network with attention mechanisms, residual functions and 3D equivariance, to predict, for each residue, its probability of belonging to the binding region of a given complex. The architecture is loosely inspired in state-of-art approaches such as Alpha Fold 3. Our model achieves an accuracy of 0.87 and an ROCauc of 0.78. The application allows for visualization of the predicted binding region for a chosen structure.

General Workflow

Data and Input Structure

The training data was gathered from PDBIND refined set of 5316 protein-ligand complexes.

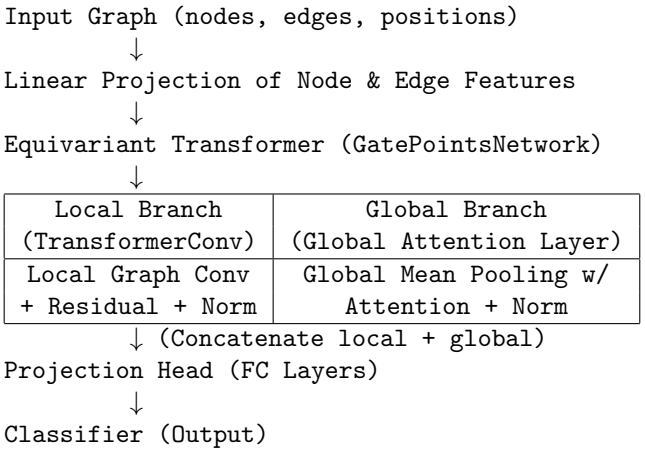
The input of the model has to go through a chain of pre-processing steps before being passed to the neural network, since the protein-ligand complex structures are described as a mathematical graph, in which the nodes represent the residues, and the edges represent spatial distances between them. The model receives the pre-processed data with 23 features per node (20 for a one-hot-encoding variable indicating the residue type, and 3 structural features).

Feature Engineering

The model has both node and edge features (remember the graph nature of the input), and uses structural information to boost its performance. The structural features in the node information are the dihedral angles (psi and phi) and the SASA (Surface accessible solvable area). These features give the model valuable information for inference, as they relate to crucial biological properties of the interaction between the residues and the whole molecules. There are also two edge features: the radial basis functions and the spherical harmonics, which help the model understand the distances between the residues in 3d space.

Model Architecture

Our model, named **Multi-scale equivariant residual network**, has the following architecture:



Output

The model outputs the probability of belonging to the binding region per each residue in the complex. We used two kinds of output: a concatenation of all probabilities per all residues in all complexes (its easier to compute performance metrics with this 1D vector of probabilities); an object with the probabilities but with the same structure/dimensions as a given complex (adequate for visualizations).

Implementation Metrics and Performance

Our model performs pretty decently, though still far from state-of-art such models with more layers and powerful computing resources. Our chosen loss function is the Binary Cross Entropy with Logits:

$$\text{BCE}(x, y) = -\left(y \cdot \log(\sigma(x)) + (1 - y) \cdot \log(1 - \sigma(x))\right)$$

where x is the logit (raw model output), $y \in \{0, 1\}$ is the ground truth label, and $\sigma(x)$ is the sigmoid function. The models goes from a BCE of 1.2 to 0.16 in 400 epochs. It uses the Adam optimizer, the most common optimizer in machine learning (akin to stochastic gradient descent but with learning rate adaptation for each parameter).

Next we can see the confusion matrix for an iteration of the model with an accuracy of around 0.87:

		Predicted	
		0	1
Actual	0	53457	5545
	1	2890	5294

Table 1: Confusion matrix.

However, just the accuracy or the error as measured by the loss function, do not give us much insight into how the model is performing. The following summary table captures more information:

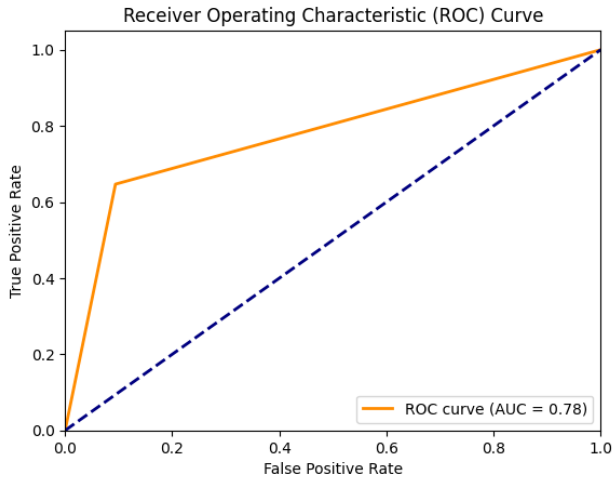
	Precision	Recall	F1-score	Support
0.0	0.95	0.91	0.93	59002
1.0	0.49	0.65	0.56	8184
Accuracy			0.87	67186
Macro avg	0.72	0.78	0.74	67186
Weighted avg	0.89	0.87	0.88	67186

Table 2: Classification report

We can see that, for the majority class (non-binding region), 95% of predicted residues were truly from non-binding regions (precision), and of all actual non-binding residues, 91% were correctly identified (recall). However, for the minority class (the one of real interest for our work), the results are far more mediocre. The precision was almost 50% and the recall was 65%. This means that of all predictions of a binding residue, half were wrong, and that 35% were false negatives (were actual binding residues).

The model was adjusted (by trial and error) to minimize false negatives over false positives, to prioritize recall for the minority class (which explains the discrepancy between precision and recall), since if the model fails to identify a true binding site, this could lead to missed drug targets or incorrect functional annotations. Anyways, false positives are also important and a convincing argument could be made either way.

To ROCauc also helps as to see the relation between the true positive rate and the false positive rate, which amounts to 0.78 in our final iteration, with much room for improvement.



Implementation details

Attention mechanism

The network uses local graph convolutions that incorporate self-attention mechanisms. This allows for local message passing weighted with the attention weights as in any usual transformer. It also incorporates a global attention layer (hence its multi-scale nature), which does global pooling with attention (not a convolution, and across the entire graph rather than with local neighbours). This

should allow the model to have information both from local and distant, more general relations between residues.

3D equivariance

Equivariance is crucial in structural bioinformatic tools such as this one, as it allows for the preservation of 3D geometric symmetries (trotation, translation, reflection) that are inherent of a given protein structure. In other words, no matter the orientation of the input complexes, since as long as their structures are the same, the model will treat them equally.

Edge features

The edge features help the model encode spatial relationships between residues by assuming the interactions between nodes depend on relative position.

To describe the pariwise position $r_{ij} = r_j - r_i$, we can decompose it in:

$$\|r_{ij}\| \in \mathbb{R}_+ \qquad \text{for distance}$$

$$\hat{r}_{ij} = \frac{r_i j}{\|r_{ij}\|} \in S^2 \qquad \text{for direction}$$

and then, we can express functions of r_{ij} using a spherical-radial basis:

$$f(\mathbf{r}_{ij}) = \sum_{\ell,m,k} c_{\ell mk}\, \phi_k(\|\mathbf{r}_{ij}\|)\, Y_\ell^m(\hat{\mathbf{r}}_{ij})$$

where $\phi(d_{ij})$ is a radial basis function of the form

$$\phi(d_{ij}) = \exp\left(-\gamma \|d_{ij} - \mu_k\|^2\right)$$

where d_{ij} is the distance between atoms i and j, and μ_k are learnable centers. And $Y_\ell^m(\mathbf{Rr})$ represents the spherical harmonics of the form

$$Y_\ell^m(\mathbf{Rr}) = \sum_{m'=-\ell}^{\ell} D^\ell_{m'm}(\mathbf{R}) Y_\ell^{m'}(\mathbf{r})$$

We consider that understanding each component of the equation is out of the scope of the present work, but as an intuition, we can think of $Y_\ell^m(\mathbf{Rr})$ as being analogous to the complex exponential basis e^{inx} in the Fourier series, but in S^2. In other words, spherical harmonics are to functions on a unit sphere what Fourier series (sines and cosines) are to functions on the unit circle. This allows the network to encode not only direction (with radial basis functions) but also in what direction things are in 3D space (interacting residues, for instance).

The spherical harmonics are used in many biophisical applications, due to the fact that they transform predictably under rotation. In our case, the features represent angular dependencies between residues, which means that the previous decomposition in radial-spherical basis is equivariant under E(3) rotations.

Imbalancement of classes

Since for any molecule in our datasets, most of the structure is not labeled as a binding region, the model could learn to predict the majority class and still perform well, albeit with no predictive power whatsoever. To avoid this possibility, we upweighted the minority class by a factor of 5, to strongly penalize minority class errors.