Structure-based protein-ligand binding prediction using deep learning

Problem statements

The availability of protein and ligand structural data has greatly advanced our understanding of protein-ligand interaction (PLI) with high resolution details. Thus, predicting protein-ligand interaction has been of great research interest, as it is significantly beneficial to discovery of new drugs which are predominantly ligands of certain proteins of biological importance. Though efforts have been made to further reveal the mechanistic relationships between structure information and protein properties, the diversity of the proteins and PILs as well as the complex quantum mechanics underlying the phenomenon is still hard to grasp, making it difficult to construct mechanistic models of PILs. In recent years, deep learning has been deployed as a proven method to predict PLIs with relatively high accuracy. In this project, we aim to develop a deep learning-based method to predict PILs from structural information.

Dataset description

Here we acquire a series of protein and ligand structures in pdb format. In the training datasets, the ground truth is defined as the protein and ligand with the same index, while the ground truth of the test datasets is not given. However, we can still construct positive samples with the training dataset. One of the examples is illustrated below on the left. Similarly, by mixing up the indice, we can create negative samples, as illustrated below on the right. Apparently, the location information may inform PIL status as illustrated.

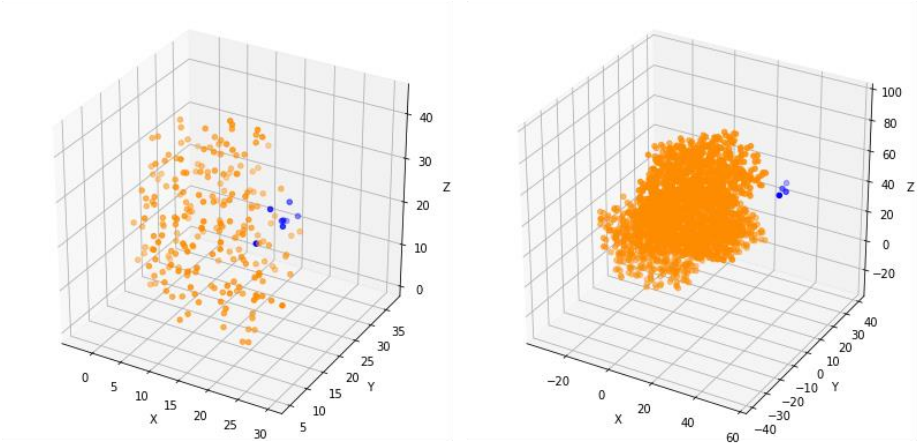


Figure 1. Visualization of a positive sample PIL (left) and a negative sample PIL (right)

Here we aim to construct such deep learning models that can select 10 best candidate ligands for each protein where the true ligand should be among the 10 candidates. We import both paired and unpaired protein-ligand complex data as positive and negative samples, respectively. The atom coordinates are voxelized to standardize the input (Figure 2). Thus, the original problem is re-structed as a 3D image classification problem, where a number of approaches, such as convolutional neural networks (CNN), are applicable (Pu et al., 2019). Each atom is annotated with their atom types, either hydrophobic or polar, which can be treated as two channels in the CNN.

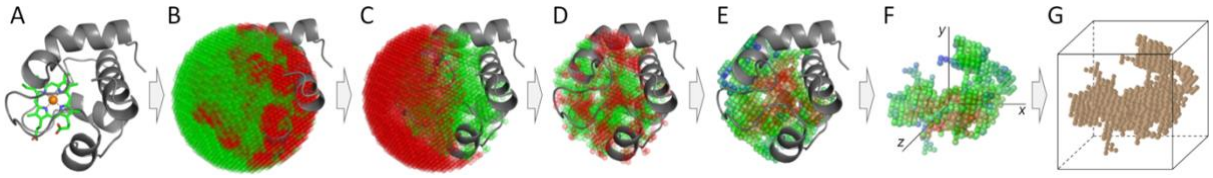


Figure 2. Illustration of voxelization. Taken from (Pu et al., 2019)

Model description

First, we construct and train a relatively simple 3D convolutional network with the structure illustrated below.

Layer (type)	Output Shape	Param #
input_1 (InputLayer)	[(None, 32, 32, 32, 2)]	0
conv3d (Conv3D)	(None, 30, 30, 30, 32)	1760
max_pooling3d (MaxPooling3D)	(None, 15, 15, 15, 32)	0
conv3d_1 (Conv3D)	(None, 13, 13, 13, 16)	13840
max_pooling3d_1 (MaxPooling3D)	(None, 6, 6, 6, 16)	0
conv3d_2 (Conv3D)	(None, 4, 4, 4, 16)	6928
max_pooling3d_2 (MaxPooling3D)	(None, 2, 2, 2, 16)	0
flatten (Flatten)	(None, 128)	0
dense (Dense)	(None, 1)	129

Total params: 22,657
Trainable params: 22,657
Non-trainable params: 0

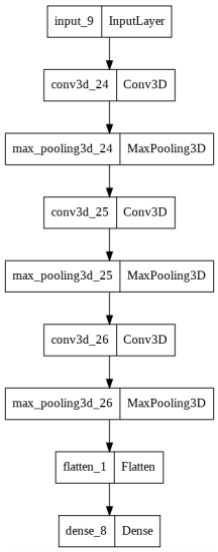


Figure 3. Model summary

Below is the training results. The results show that although training loss declines over time, testing loss increases over time, which suggests overfitting. The model accuracy in training data approches 100% yet in testing data, the accuracy fluctuates at around 92%. The code is available at [1]. The prediction results have been attached in the project folder.

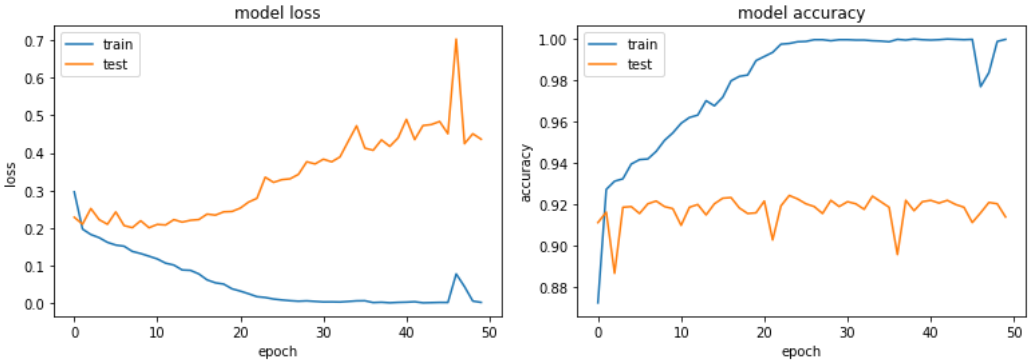


Figure 4. Model performance

[1] https://colab.research.google.com/drive/1Xnl_qDcdLQgDOvqDi_H6yxTncM1MKmqC?usp=sharing

Additionally, we reuse the code from the DeepPL project at [2], which was designed for this problem. The structure of the model is shown below:

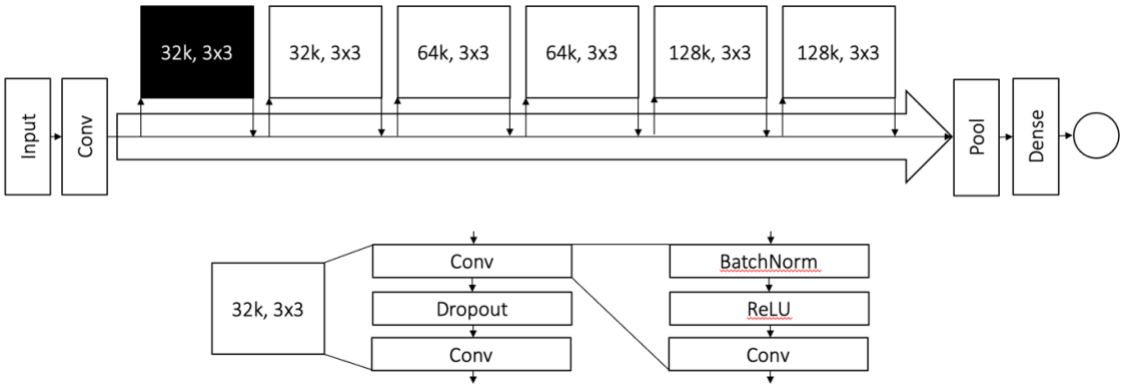
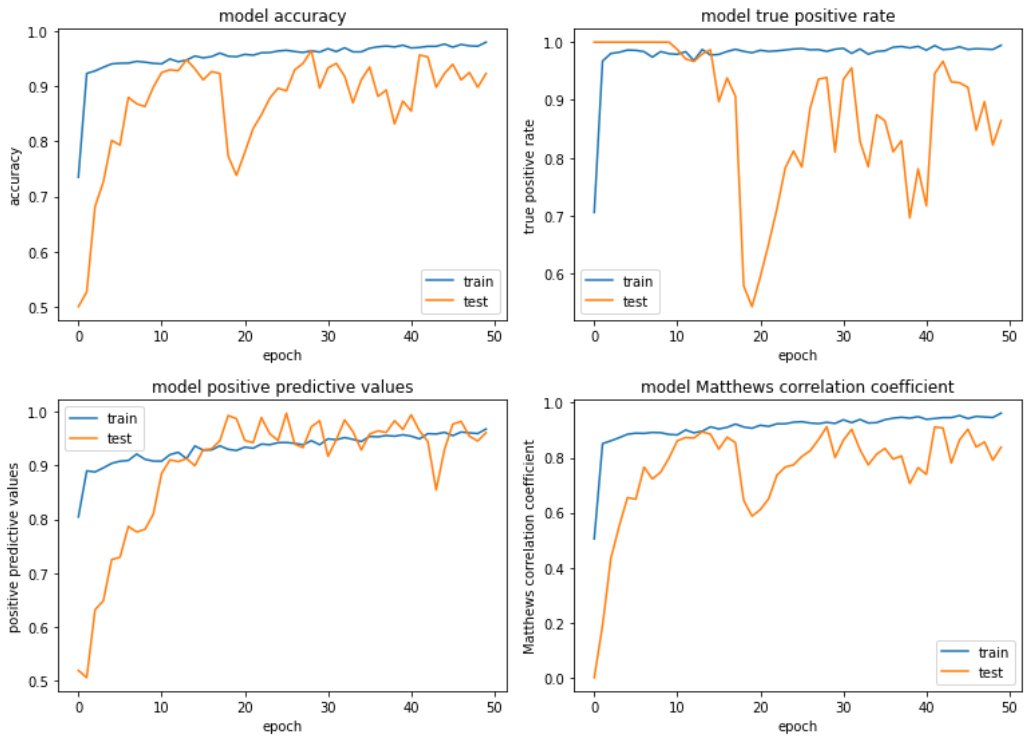


Figure 4. Model summary of wide resnet based model. Taken from [3].

The author introduces wide residual network (Zagoruyko & Komodakis, 2016) to improve training performance. We performed the training with Google Colab at [4], with modifications to make it work with the current version of TensorFlow. Below is our training result. The prediction results have also been attached.



[2] https://github.com/FausticSun/DeepPL
[3] https://github.com/FausticSun/DeepPL
[4] https://colab.research.google.com/drive/1muPDvTf9158Z5TIclLVVUqoqwQewUDNG?usp=sharing

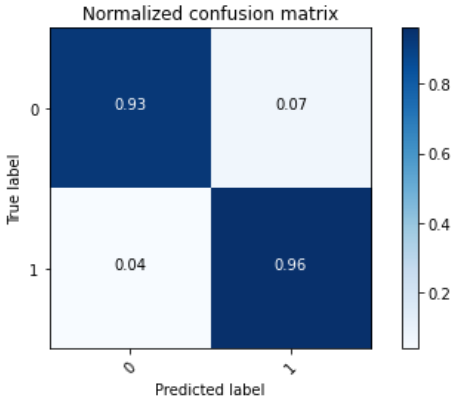


Figure 5. Model performance of wide resnet based model

The test accuracy improves to around 94%, which is consistent with other metrics such as true positive rates, positive predictive values, and Matthew's correlation coefficients. Although both models can achieve relatively high accuracy, the latter is much faster to train, which halves the training time given the same epoch number, and it has better prediction performance as indicated by best validation accuracy. This could be thanks to the optimal network structure that avoid diminished gradients.

Future directions

Since the training is extremely time-consuming, I didn't do much parameter tuning with regard to learning rate, cube size, etc. The model structure of my original model is as simple as possible, where batch normalization layers could be added to make training faster. However, mathematically, it still produces a lot of false positive results given the number of proteins as high as 824. Thus, further fine-tuning of the model could improve the model performance and prediction accuracy. Apart from the 3D image classification-like approaches we use, there are also graph-based methods suggested by (Li et al., 2021), which may better represent the atom type data and may be worth exploring in the future.

References

Li, S., Zhou, J., Xu, T., Huang, L., Wang, F., Xiong, H., Huang, W., Dou, D., & Xiong, H. (2021). *Structure-aware Interactive Graph Neural Networks for the Prediction of Protein-Ligand Binding Affinity*. http://arxiv.org/abs/2107.10670

Pu, L., Govindaraj, R. G., Lemoine, J. M., Wu, H.-C., & Brylinski, M. (2019). DeepDrug3D: Classification of ligand-binding pockets in proteins with a convolutional neural network. *PLOS Computational Biology*, *15*(2), e1006718. https://doi.org/10.1371/journal.pcbi.1006718

Zagoruyko, S., & Komodakis, N. (2016). Wide residual networks. *ArXiv Preprint ArXiv:1605.07146*.