BiRDS - Binding Residue Detection from Protein Sequences using Deep ResNets

Vineeth R. Chelur and U. Deva Priyakumar*

Center for Computational Natural Sciences & Bioinformatics

International Institute of Information Technology

Hyderabad - 500032, India

E-mail: deva@iiit.ac.in

Abstract

Protein-drug interactions play important roles in many biological processes and therapeutics. Prediction of the active binding site of a protein helps discover and optimise these interactions leading to the design of better ligand molecules. The tertiary structure of a protein determines the binding sites available to the drug molecule. A quick and accurate prediction of the binding site from sequence alone without utilising the three-dimensional structure is challenging. Deep Learning has been used in a variety of biochemical tasks and has been hugely successful. In this paper, a Residual Neural Network (leveraging skip connections) is implemented to predict a protein's most active binding site. An Annotated Database of Druggable Binding Sites from the Protein DataBank, sc-PDB, is used for training the network. Features extracted from the Multiple Sequence Alignments (MSAs) of the protein generated using DeepMSA, such as Position-Specific Scoring Matrix (PSSM), Secondary Structure (SS3), and Relative Solvent Accessibility (RSA), are provided as input to the network. A weighted binary cross-entropy loss function is used to counter the substantial imbalance in the

1

two classes of binding and non-binding residues. The network performs very well on single-chain proteins, providing a pocket that has good interactions with a ligand.

Introduction

The rapid speed of sequencing attained with modern DNA sequencing technology has been instrumental in sequencing complete DNA sequences, which leads to faster sequencing of proteins. Although there have been improvements in the determination of the three-dimensional protein structure by techniques such as X-ray Crystallography, NMR Spectroscopy and Cryo-Electron Microscopy, the gap between the number of known protein sequences (214,406,399 UniProt sequences as of May 2021),[1] and the number of known structures (177,910 PDBs as of May 2021)[2][3] is large. Proteins perform a vast array of functions within organisms, and the tertiary structure of a protein can provide important clues about these functions.

In drug design, new drugs are designed based on a biological target such as a protein. Identification of the potential active binding site of a protein is an essential step in drug design. Predicting the binding site of a protein, based on the amino acid sequence is a challenging problem and is helpful to identify potential binding residues before the three-dimensional structure becomes available. Ligand binding site prediction methods can be classified into four groups: 3D structure-based, template similarity-based, traditional machine learning-based and deep learning-based prediction methods.

Three-dimensional structure-based methods assume that most small ligand bindings occur in cavities on protein surfaces because large interfaces have a high affinity. Hence, these methods locate the binding site by searching for spatial geometry or energy features (by placing probes) in protein structures. SITEHOUND[4] uses a carbon and phosphate probe inside a grid that covers the entire protein. The grid points with higher interaction energies are clustered and determine the binding residues. CURPOCKET[5] is a spatial geometric measurement method that computes the curvature distribution of the protein surface and

identifies clusters of concave regions. Some other methods include CASTp,[6] LIGSITE,[7] VIS-CANA,[8] Fpocket,[9] Patch-Surfer2.0.[10] While 3D structure-based methods have been widely used, they depend on various factors such as the resolution of the structure determination method, the presence or absence of ligand groups, and external molecules.

Template similarity-based methods do not consider proteins as independent entities but evolved from structurally, functionally or sequentially similar proteins. S-SITE and TM-SITE[11] employ the Needleman-Wunsch algorithm to align the query protein to each of the proteins in the BioLip[12] database and selects similar sequences according to the alignment. The binding residues of the aligned proteins, which occur more frequently, are considered the binding site. Methods such as ConSurf,[13] FINDSITE,[14] 3DLigandSite,[15] FunFOLD,[16] COFACTOR,[17] employ template-similarity.

Traditional machine learning-based methods build an analytical model based on protein data to identify patterns and structural similarities. 3D structure-based and template similarity-based methods complement each other well. Machine learning is used to integrate the information of both methods and apply mathematical functions to improve prediction accuracy. P2RANK[18][19] uses a random forest algorithm to predict ligandibility scores (the ability of a ligand to bind to specific points on the protein) across the entire protein surface. The points with high scores are then clustered into a single binding pocket. Many methods have started using Machine Learning in the recent past. ConCavity,[20] MetaPocket,[21] RF-Score,[22] NsitePred,[23] NNSCORE[24],[25] LigandRFs,[26] COACH-D[27] and Taba[28] are some of them.

Deep Learning is a subfield of machine learning based on artificial neural networks with feature learning. When a deep learning network is fed large amounts of data, it can automatically discover the representations needed for feature detection or classification. Deep learning has been hugely successful in the general areas of drug design such as binding affinity predictions,[29,30] protein contact map predictions,[31,32] and protein-structure predictions.[33–35] Deep learning-based methods like DeepSite,[36] and Kalasanty[37] model binding site predic-

tion as an image processing problem. They voxelize the protein 3D structure into small grids and calculate the specific properties of each grid. These values are then used to train a deep convolutional neural network that predicts whether a grid belongs to a binding site. DeepCSeqSite[38] is a template-based method that uses seven characteristics (position-specific scoring matrix, relative solvent accessibility, secondary structure, dihedral angle, conservation scores, residue type and positional embeddings) of each residue to create a feature map, which is then used as an input to a convolutional neural network. DeepPocket[39] is a structure-based method that uses 3D Convolutional Neural Networks to generate a list of pocket probabilities and a segmentation model to elucidate shapes for the top-ranked pockets.

In this paper, a deep residual neural network (ResNet)[40] is trained to make the binary prediction of whether an amino acid residue of the sequence belongs to the primary binding site or not (BiRDS). The Multiple Sequence Alignment (MSA) of the protein sequence is calculated, and the feature map is extracted from the MSAs. BiRDS is trained on the feature map of all proteins in the training dataset, using a weighted binary cross-entropy loss. The network outputs the final probabilities, which are converted to binary outputs. The network does very well in recognising the binding sites of individual protein chains.

Methods

Dataset

For the training and validation of the model, the sc-PDB[41] dataset (v. 2017) is used. The database takes samples from the Protein Data bank[2,42] and creates prepared protein structures and the most ligandable binding site. Thus each sample in the dataset contains the three-dimensional structure of one ligand, one protein, and one site, all stored in mol2 format.

4

Typically, the complete structure of a protein is unavailable due to missing residues, and hence the entire sequence of the protein is obtained from RCSB[3] website*. A one-to-one mapping of the amino acids in the sequence to the amino acids in the protein mol2 file is required to know which amino acid belongs to a binding site. This mapping is done by first extracting the protein sequence from the 3D structure. Next, the Needleman-Wunsch dynamic programming algorithm[43] is used to align the sequence extracted from the structure file to the downloaded RCSB sequence. This algorithm is implemented using a modified version of Zhanglab's NW-Align program.[44] The protein structure file is then reindexed, based on this alignment, to match the indexing of the RCSB sequence. This way, the specific binding residues can be labelled in the RCSB sequence.

The training set consists of 17,594 PDB structures with 28,959 sequences (9519 unique sequences), originating from 1240 organisms, with the most abundant being humans(28.26%). The dataset was diverse and contained proteins from 1996 different PFAM families and 856 PFAM clans.

Table 1: Summary of the dataset used for training and testing

	N_{prot}	N_{br}	N_{nbr}	$P_{br}(\%)$
Train	15,860	589,329	8,725,043	6.33
Test	2,464	86,230	1,345,646	6.02

N_{prot} - Number of proteins N_{br} - Total number of binding residues

N_{nbr} - Total number of non-binding residues $P_{br}(\%)$ - Percentage of binding residues

This data is split into 10-folds (each containing 1586 structures), based on Uniprot ID, precisely like how Stepniewska-Dziubinska et al. did in their study.[37] This split ensures no data leakage between the validation and training set by putting all structures of a single protein in the same fold.

A separate test set, SC6K, is constructed using all PDBs from 2018 onwards, till 28th February 2020. All PDBs having at least one ligand, during this period, are run through

---

pdbconv from the IChem Toolkit[45] to generate a dataset with the same filters and site selection as sc-PDB.[41] The test set consists of 2,274 PDB structures with 3,434 sequences (1889 unique sequences), originating from 548 organisms, with the most abundant being humans (23.76%). The test set contained proteins from 882 PFAM families and 452 PFAM clans.

Features

There are 9519 unique protein sequences in the sc-PDB[41] dataset and 1889 unique protein sequences in the test set. The MSAs are generated for these sequences using the method described below and stored in PSICOV[46] .aln format. The features are similar to the ones used by DeepCSeqSite[38] and are commonly used in sequence-based predictions: One-hot encoding and Positional embeddings are extracted from the sequence alone. Position Specific Scoring Matrix, Information Content, Secondary Structure and Solvent Accessibility are extracted from the generated high-quality MSAs.
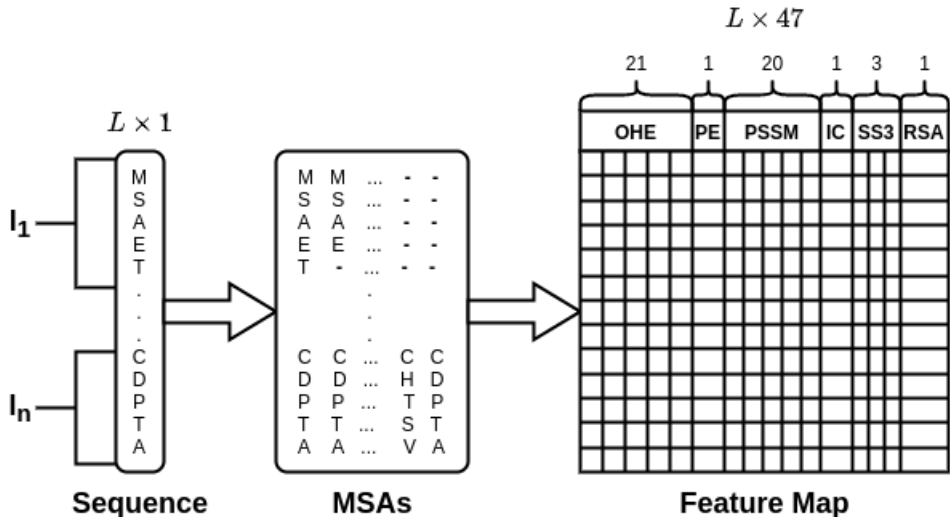

Figure 1: Process used for generating the feature map of BiRDS framework. The MSAs of the individual chains of a protein are generated using DeepMSA, which are used to generate the features of the protein chains. These features are concatenated to form the feature map

MSA Generation

Collections of multiple homologous sequences (called Multiple Sequence Alignments or MSAs) can provide critical information for modelling the structure and function of unknown proteins. DeepMSA[47] is an open-source method for sensitive MSA construction, which has homologous sequences and alignments created from multiple sources of databases through complementary hidden Markov model algorithms.

The search is done in 2 stages. In stage 1, the query sequence is searched against the UniClust30[48] database using HHBlits from HH-suite[49] (v2.0.16). If the number of effective sequences is < 128, Stage 2 is performed where the query sequence is searched against the Uniref50[50] database using JackHMMER from HMMER[51] (v3.1b2). Full-length sequences are extracted from the JackHMMER raw hits and converted into a custom HHBlits format database. HHBlits is applied to jump-start the search from Stage 1 sequence MSA against this custom database.

One-Hot Encoding and Positional Embeddings

There are 21 amino acids in the vocabulary, 20 standard (labelled in alphabetical order from 1 to 20), and X (labelled 0, representing non-standard amino acids). The one-hot encoding (OHE) of an amino acid will be a vector of zeroes of length 21, where the position of the amino acid in the vocabulary is marked with a one. OHE is used to help the model differentiate between the different types of amino acids. Positional Embeddings (PE) carry information about the absolute position of the amino acids in the sequence. A simple method of embedding is used where the position of the jth amino acid is represented by $PE_j = \frac{j}{l_i}$, where l_i is the length of the ith chain of the protein.

Position Specific Scoring Matrix and Information Content

Position Specific Scoring Matrix (PSSM) is a commonly used representation of patterns in biological sequences. PSSMs are derived from MSAs using Easel[52] and Heinikoff position-based weights so that similar sequences collectively contributed less to PSSM probabilities than diverse sequences. The information content (IC) of a PSSM gives an idea about how different the PSSM is from a uniform distribution. IC is also derived using Easel.

Secondary Structure and Solvent Accessibility

The secondary structure is defined by the pattern of hydrogen bonds formed between the amino hydrogen and carboxyl oxygen atoms in the peptide backbone. It gives an idea of the three-dimensional structure of the protein. The secondary structural elements are alpha helices, beta sheets and turns. PSIPRED (v4.0)[53] is used to predict the probability of each state of the 3-state secondary structure (SS3) for every amino acid in the sequence. The solvent-accessible surface area is the surface area of a biomolecule that is accessible to a solvent. SOLVPRED from MetaPSICOV 2.0[54] is used to predict the relative solvent accessibility (RSA) of every amino acid in the sequence. RSA can be calculated as $RSA = ASA/MaxASA$, where ASA is the solvent-accessible surface area, and MaxASA is the maximum possible solvent accessible surface area for the amino acid residue.

Model

BiRDS Architecture

A Convolutional Neural Network (CNN) is a Deep Learning algorithm that can take an image as input, assign importance (learnable weights and biases) to various aspects/objects in the image, and differentiate one from the other. When multiple CNN layers are stacked on top of each other, Deep Neural Networks (DNNs) are formed. DNNs are difficult to train because of the vanishing gradient problem where the gradients become so small that the

network's weights do not change, preventing further training. With the introduction of skip connections (shortcuts to jump over some layers) in CNNs, the vanishing gradient problem is avoided. CNNs with skip connections are known as Residual Neural Networks or ResNets.[40] ResNets use representation learning to extract the most important features for classification. They can also model long-range interactions very well and hence have been very successful in the field of Computational Natural Sciences.[33] The architecture of the deep Residual Neural Network used here is shown in Figure 2.

Each sample protein in the dataset consists of one or more protein sequences. Let the length of the sequences be $l_1, ..., l_n$. Features are generated for each sequence in the protein (ordered by chain ID in PDB), leading to multiple vectors of shape $[l_i, 47]$ for the ith sequence. These generated features are combined through simple concatenation, giving a final feature vector of shape $[L, 47]$ as input to the model, where $L = l_1 + ... + l_n$.
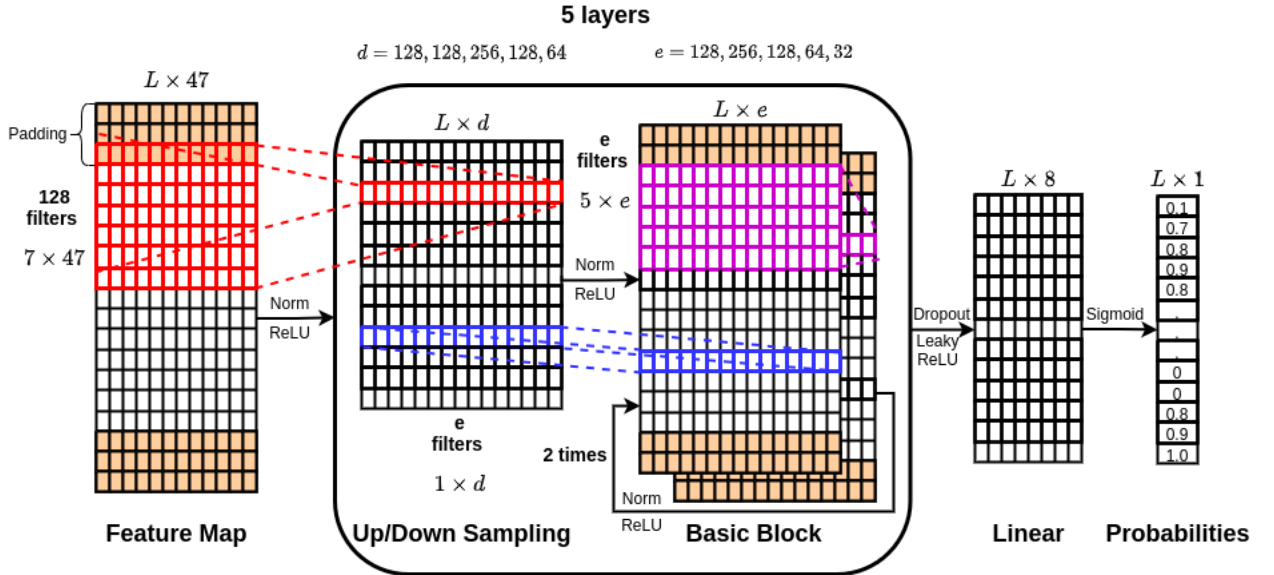

Figure 2: Architecture of the deep learning model, BiRDS

The feature vector is passed through the first level, consisting of a 1D convolutional layer with 128 filters, each filter size being 7, a batch normalisation layer and the ReLU (Rectified Linear Unit) activation function. The input is padded with zeroes to ensure that the length of the output vector remains the same. The filters of the convolution layer stride across

9

the length of the protein, considering the features of the three amino acids before, the three amino acids after and the current amino acid (totalling 7). This stride along the input allows for the extraction of the required information of the current amino acid based on the features of nearby amino acids.

The following few levels consist of 2 blocks called BasicBlocks. A BasicBlock consists of a 1D convolutional layer, a batch normalisation layer, a ReLU activation function, a second 1D convolutional layer, a second batch normalisation layer, and a final ReLU activation. The skip connection is made after the final ReLU activation, where the initial input to the BasicBlock is added to the output of the final ReLU activation. Usually, the input and output size of the first BasicBlock do not match, and hence there is an up/down-sampling layer that ensures that the input has the same shape as that of the output. The output from the previous level passes through the first BasicBlock, which has an up/down-sampling layer and then goes through the second BasicBlock. In the proposed architecture, the number of filters used at each level goes from $128 \rightarrow 256 \rightarrow 128 \rightarrow 64 \rightarrow 32$, with the filter size being 5 for every convolution.

The last two levels contain a simple, linear, fully connected artificial neural network. The last but one level has a LeakyReLU activation function along with a dropout as well. The last level has a Sigmoid function to ensure that the model's output is between $[0, 1]$. The final output of the model is a vector of size L (length of the protein), denoting the probabilities of a residue being a part of the binding site.

Loss Function

For model training, the loss function is a weighted binary cross-entropy and is given by $L(\hat{y}, y) = -(\alpha \hat{y} \log(y) + (1 - \hat{y}) \log(1 - y))$, where $\hat{y}$ is the vector of true labels of whether an amino acid belongs to the binding site or not, y is the model output of probabilities of a residue belonging to a binding site, α is the weight that is assigned to the rare class.

The main problem in this classification task is the substantial imbalance in the two classes of binding and non-binding residues. As shown in Table 1, the percentage of binding residues is only around 6%. Hence, α is used to penalise the model more heavily if it incorrectly predicts binding residues. α is calculated on the fly for every batch of inputs as $\alpha = \frac{n_{nbr}}{n_{br}}$, where n_{nbr} is the total number of non-binding residues in the batch and n_{br} is the total number of binding residues in the batch.

Evaluation Metrics

Confusion Matrix

A confusion matrix is a table that allows for the visualisation of the performance of a supervised learning algorithm. The following terminologies can be defined in the binary classification of a residue as a binding residue (BR) or non-binding residue (NBR).

- True Positive (TP): Number of BRs predicted correctly as BRs.

- True Negative (TN): Number of NBRs predicted correctly as NBRs.

- False Positive (FP): Number of NBRs predicted incorrectly as BRs.

- False Negative (FN): Number of BRs predicted incorrectly as NBRs.

The following metrics can be derived from the confusion matrix

Accuracy: $ACC = \frac{TP+TN}{TP+TN+FP+FN}$

Precision: $PPV = \frac{TP}{TP+FP}$

Recall: $TPR = \frac{TP}{TP+FN}$

F1 score: $F_1 = \frac{2TP}{2TP+FP+FN}$

Matthews Correlation Coefficient: $MCC = \frac{TP \times TN - FP \times FN}{\sqrt{(TP+FP)(TP+FN)(TN+FP)(TN+FN)}}$

MCC

The Matthew's Correlation varies from $[-1, +1]$, with $+1$ representing a perfect prediction, 0 representing no better than a random prediction and -1 representing total disagreement between the prediction and the observation.

DCC

DCC is the distance between the centre of the predicted binding pocket and the centre of the actual binding pocket. It is commonly used for evaluating 3D-structure based models. The success rate of DCC is defined as the fraction of predictions below a given threshold. Predicted pockets with DCC below 4Å are considered to be correctly located.

Implementation

The model is implemented using PyTorch Lightning,[55] which is a wrapper on the popular open-source deep-learning library, PyTorch.[56] The model is trained in batches using an Adam Optimizer with the ReduceLROnPlateau scheduler, maximising the MCC on the validation set. A learning rate warm-up is used, where the learning rate is gradually increased to the actual learning rate during the first epoch. The implementation can be found at https://github.com/devalab/BiRDS.

Results and Discussion

The sc-PDB[41] dataset was split into ten folds, and ten models with the same architecture were trained. One fold formed the validation set, and the remaining folds formed the training set for each of the models. The validation results are provided in Table 2, along with the confusion matrix in Figure 3

Table 2: Validation results of all 10 trained models and test results

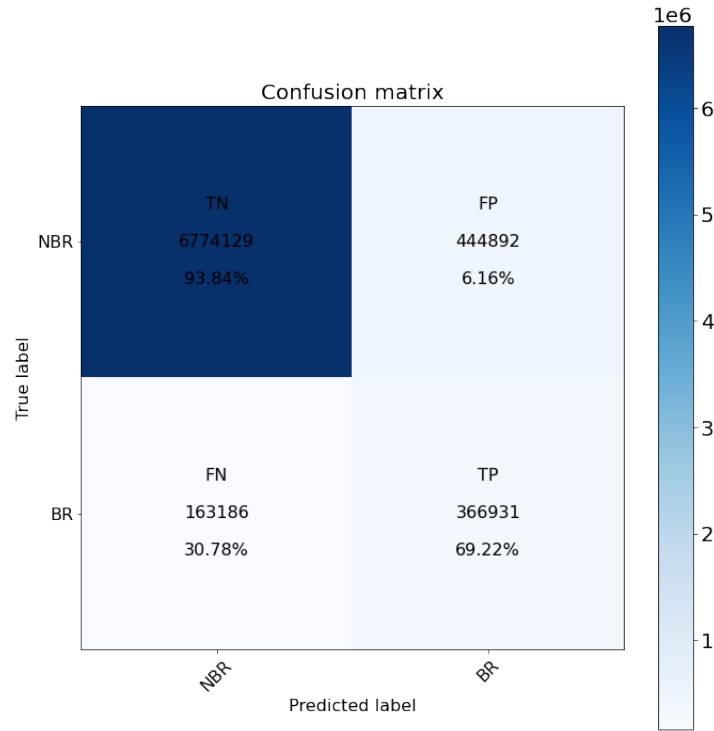| Dataset	ACC(%)	PPV(%)	TPR(%)	F1(%)	MCC(%)
Fold 1	92.58	48.64	70.65	57.62	54.83
Fold 2	92.18	46.80	67.29	55.20	52.08
Fold 3	92.94	44.85	69.37	54.48	52.27
Fold 4	91.28	39.62	65.73	49.44	46.72
Fold 5	91.74	46.33	73.11	56.72	54.07
Fold 6	92.19	47.16	69.70	56.25	53.34
Fold 7	91.90	45.45	69.55	54.98	52.13
Fold 8	92.52	47.58	68.16	56.04	53.10
Fold 9	92.06	41.86	69.69	52.31	50.14
Fold 10	92.08	44.54	68.88	54.10	51.41
Test	94.05	50.46	67.45	57.73	55.27

Figure 3: Sum of confusion matrices of the 10 models on their corresponding validation set

For testing, the ten trained models are run on the test set. An amino acid belongs to the binding site for the consensus algorithm if five or more models predict the same. The test results are also provided in Table 2, along with the confusion matrix in Figure 4
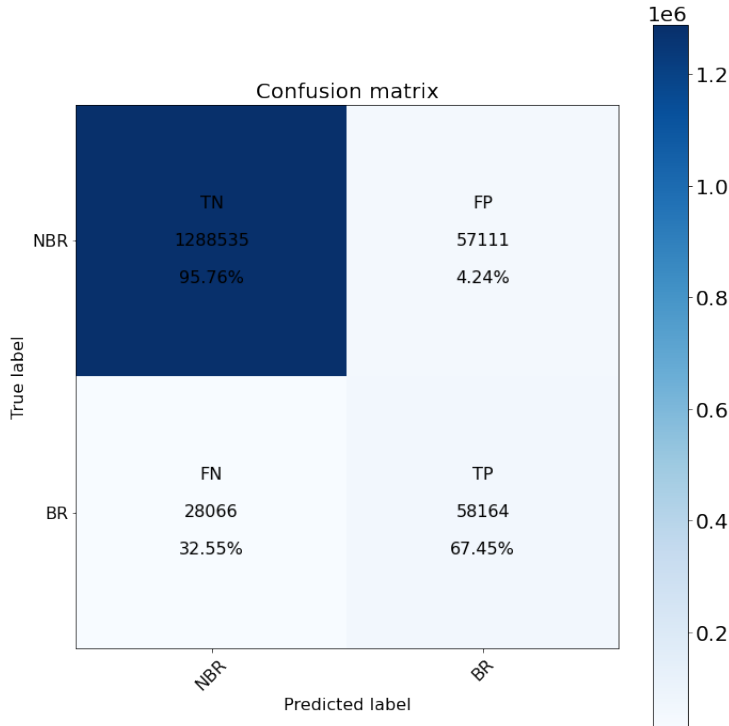

Figure 4: Confusion matrix on the test set after averaging the predictions of the 10 models

The model predictions were mapped back to the available 3D structures of proteins for the calculation of DCC. Figure 5 denotes the cross-validation results. The deep learning model is the same across all ten splits of training and validation datasets. The success rate of the models varies based on the fold that is used for validation. It ranges from 33% to 49% success rate when the DCC threshold is less than 4Å. Figure 6 denotes the test result. The predictions have a 40% success rate when the DCC threshold is less than 4Å, meaning that for 40% of the test data, the model has predicted the binding site such that the centre of the predicted binding site is within 4Å of the centre of the true binding site. As the threshold of DCC increases, the success rate also naturally increases. One should note that even if the model predicts the whole binding site correctly and misses out on a couple of residues or predicts more residues, the centre of the predicted binding site will change significantly.
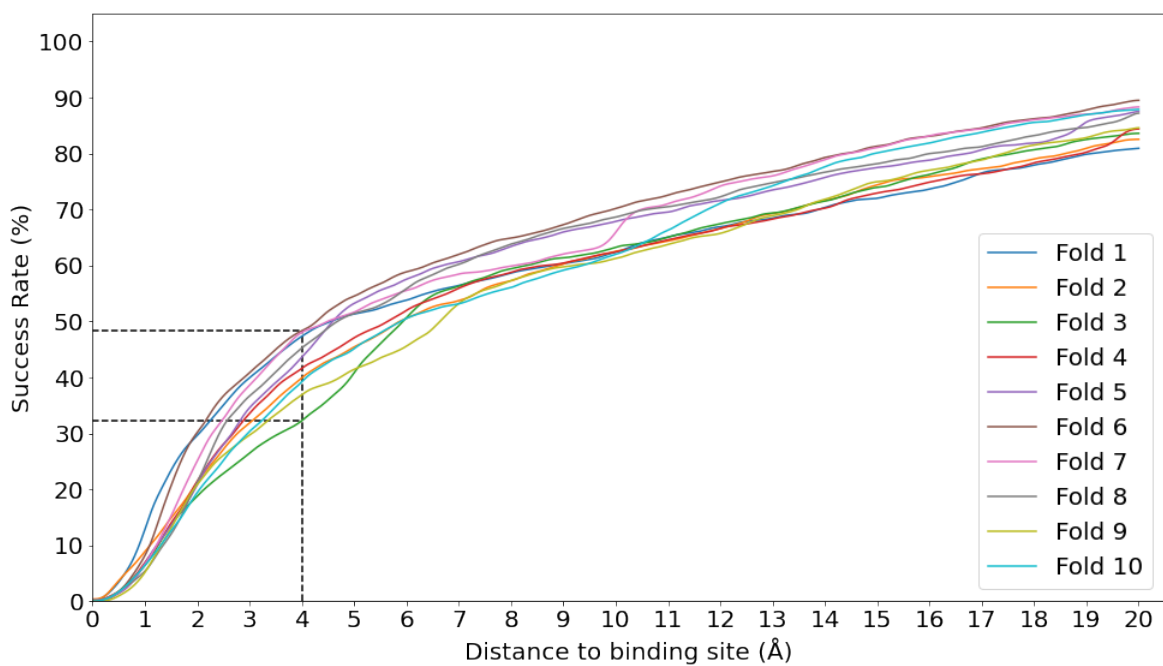
Figure 5: Success rate plot for various DCC thresholds of the 10 models on their corresponding validation set
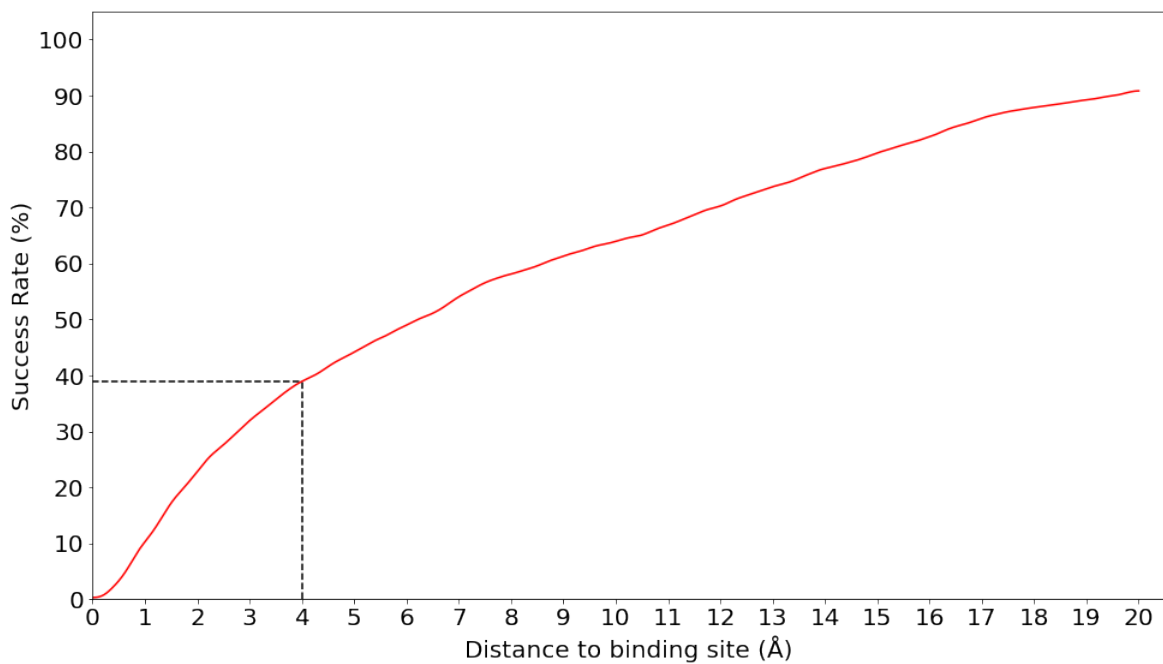

Figure 6: Success rate plot for various DCC thresholds on the test set after averaging the predictions of the 10 models

On their test set, DeepSite, a 3D structure-based deep learning model[36] achieved around 40% success rate. Given that our model uses only the amino acid sequence information, its performance is commendable. Even though the test sets are not the same, it still gives an idea of how well the current model performs.

All the currently available methods for predicting the binding site of a protein, based on sequence, predict the site only for a specific set of ligands, while our model predicts the most ligandable binding site irrespective of the ligand. Hence, there is no available method of comparison with our model for the sequence-based prediction of binding residues on the sc-PDB[41] dataset.

As a simple way to test the effectiveness of our model against DeepCSeqSite's model, we followed two approaches:

1. Run the trained DeepCSeqSite model on our test set

2. Reimplement DeepCSeqSite model architecture, train on our dataset, and then test

Both approaches failed to provide good results, with approach one providing an MCC score of 0.05 and approach two providing an MCC score of 0.1. The failure in approach one could be due to the difference in the method for generating the MSAs.

A variety of deep learning models such as Complementary GANs, stacked BiLSTMs, and TAPE protein embeddings were used for training as a means to improve the predictions. Features, such as backbone angles, secondary structure, solvent accessibility and contact number, generated using SPOT-1D[31] were also tried as features. None of them provided any significant improvement over vanilla ResNets and simple features. Instead, they added unnecessary, additional computational costs.

Some case studies were undertaken to show that the model performs as expected, but the metrics do not rate it well due to limitations of the dataset. The aggregated predictions of the models on the test set were mapped back to the three-dimensional structure of the protein-ligand complex to see how good the predictions are. The 3-D structures were

generated using 3Dmol.js.[57] In the following examples, the colour red indicates an incorrect prediction of the amino acid as a binding residue. Blue indicates an amino acid that is a binding residue but was not predicted as binding by the model. Green indicates an amino acid that was correctly predicted as binding.

In Figure 7, it looks like the model is mispredicting everything for 5YMX,[58] but it is predicting another binding site of the protein. The sc-PDB[41] dataset was generated through a series of filters, and the residues surrounding the most buried ligand was selected to be the most ligandable binding site. This selection, unfortunately, is a flaw of the dataset and the method used for predictions. There is no right way to cover cases like these where the model needs to be penalised less when it predicts a binding site that is not the most ligandable binding site. Hence, the evaluation metrics used will generally give an abysmal score for such cases.
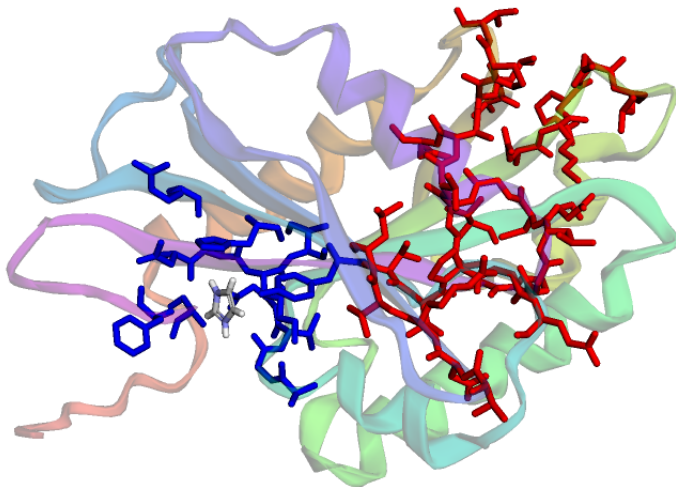

Figure 7: 5YMX - The model seems to be mispredicting the actual binding site (in blue), but is in fact, predicting another binding site of the protein (in red)

Figure 8 shows 6HU9,[59] where the model predicts individual binding sites of two proteins with the same sequence, but it finds it difficult to predict the binding site created due to the interaction between the two proteins. This may be due to the way the features are generated. It is not easy to generate features of the protein since the MSAs are generated for individual chains of a protein, not providing any information about the interaction between the chains.

Due to the fewer samples containing a binding site between the interaction of protein chains, the model finds it difficult to learn the same.
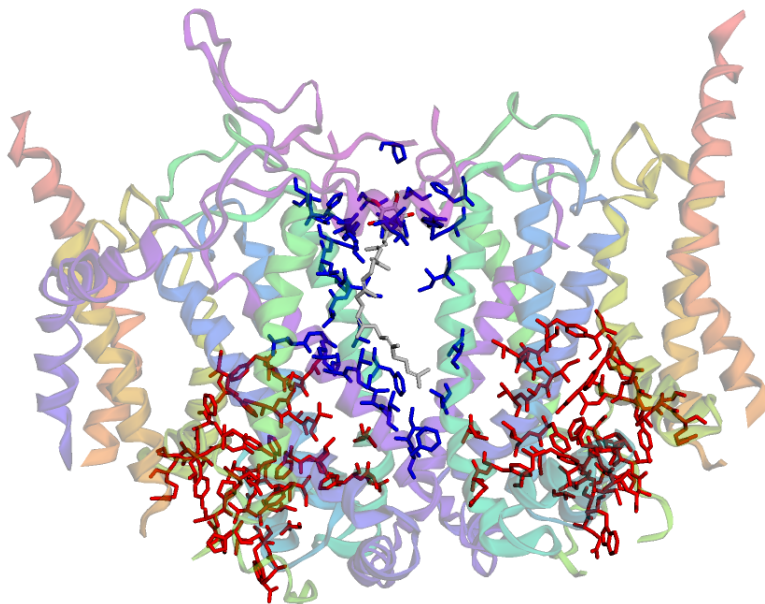

Figure 8: 6HU9 - The model is able to predict the individual binding site of the protein (in red), but not the interaction binding site (in blue)

Figure 9 shows 6PF6,[60] where the model predicts the binding site with good accuracy. It can be seen that it predicts most of the residues around the binding site and also a couple more just outside the binding site. Since the two chains of the protein have the same sequence, the model predicts identical binding residues for both chains. sc-PDB selects only one binding site as the most active binding site, and therefore, only that site is used for calculating the metrics. The metrics do not do justice to these types of predictions and score the model very poorly, even though it is doing very well.
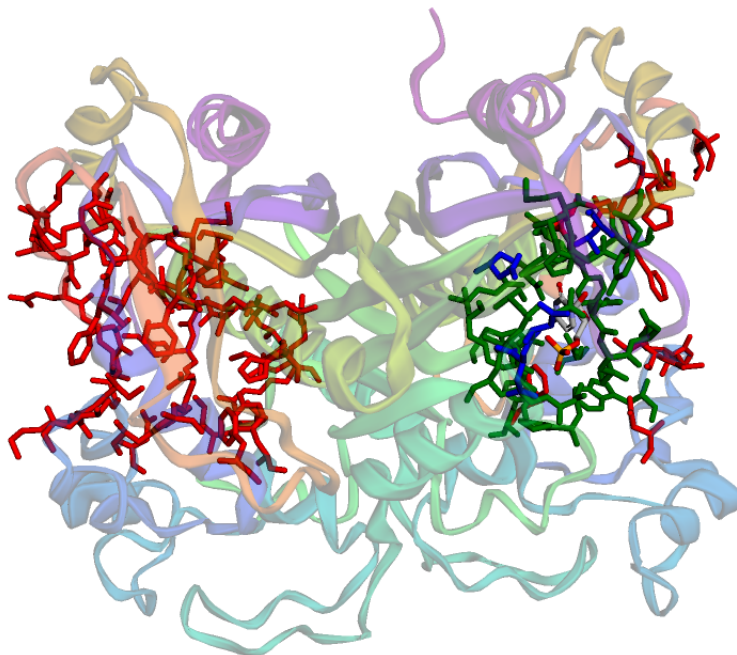
Figure 9: 6PF6 - The model predicts the binding site correctly, but due to the presence of the same chains in the protein, it predicts both the binding sites (in green and red)

Conclusion

In this study, a ResNet was used to predict the most active binding site of a protein. The sc-PDB[41] dataset contained data of the most ligandable binding site of a protein. MSAs were generated for all protein chains in the dataset using DeepMSA, and features such as Position-Specific Scoring Matrix, Secondary Structure and Solvent Accessibility were calculated. A new test set, SC6K, considering proteins in the last three years, was made using the same filters used to create the sc-PDB dataset. The individual features of the protein chains were concatenated, and BiRDS was trained using 10-fold cross-validation. BiRDS can predict the binding site of a protein fairly accurately using only the sequence information, although it fails to predict the binding site formed between protein chains due to limited data. In drug design, it becomes crucial to determine the pocket where the drug molecule binds with the protein. BiRDS (our method) can be used for early and quick determination of the binding site before the availability of the protein structure.

Data and Software Availability

The method implementation, data and pretrained models can be found at https://github.com/devalab/BiRDS.

Acknowledgement

Supporting Information Available

Supporting Information contains the resolved PDB IDs of the obsoleted PDBs of the train set and PDB IDs removed from the test set, SC6K, due to preprocessing errors.

References

(1) Consortium, T. U. UniProt: the universal protein knowledgebase in 2021. *Nucleic Acids Research* **2020**, *49*, D480–D489.

(2) Berman, H. M.; Westbrook, J.; Feng, Z.; Gilliland, G.; Bhat, T. N.; Weissig, H.; Shindyalov, I. N.; Bourne, P. E. The protein data bank. *Nucleic acids research* **2000**, *28*, 235–242.

(3) Burley, S. K.; Bhikadiya, C.; Bi, C.; Bittrich, S.; Chen, L.; Crichlow, G. V.; Christie, C. H.; Dalenberg, K.; Di Costanzo, L.; Duarte, J. M., et al. RCSB Protein Data Bank: powerful new tools for exploring 3D structures of biological macromolecules for basic and applied research and education in fundamental biology, biomedicine, biotech-

nology, bioengineering and energy sciences. *Nucleic acids research* **2021**, *49*, D437–D451.

(4) Hernandez, M.; Ghersi, D.; Sanchez, R. SITEHOUND-web: a server for ligand binding site identification in protein structures. *Nucleic acids research* **2009**, *37*, W413–W416.

(5) Liu, Y.; Grimm, M.; Dai, W.-t.; Hou, M.-c.; Xiao, Z.-X.; Cao, Y. CB-Dock: a web server for cavity detection-guided protein–ligand blind docking. *Acta Pharmacologica Sinica* **2020**, *41*, 138–144.

(6) Dundas, J.; Ouyang, Z.; Tseng, J.; Binkowski, A.; Turpaz, Y.; Liang, J. CASTp: computed atlas of surface topography of proteins with structural and topographical mapping of functionally annotated residues. *Nucleic acids research* **2006**, *34*, W116–W118.

(7) Hendlich, M.; Rippmann, F.; Barnickel, G. LIGSITE: automatic and efficient detection of potential small molecule-binding sites in proteins. *Journal of Molecular Graphics and Modelling* **1997**, *15*, 359–363.

(8) Amari, S.; Aizawa, M.; Zhang, J.; Fukuzawa, K.; Mochizuki, Y.; Iwasawa, Y.; Nakata, K.; Chuman, H.; Nakano, T. VISCANA: visualized cluster analysis of protein-ligand interaction based on the ab initio fragment molecular orbital method for virtual ligand screening. *Journal of Chemical Information and modeling* **2006**, *46*, 221–230.

(9) Le Guilloux, V.; Schmidtke, P.; Tuffery, P. Fpocket: an open source platform for ligand pocket detection. *BMC bioinformatics* **2009**, *10*, 1–11.

(10) Zhu, X.; Xiong, Y.; Kihara, D. Large-scale binding ligand prediction by improved patch-based method Patch-Surfer2. 0. *Bioinformatics* **2015**, *31*, 707–713.

(11) Yang, J.; Roy, A.; Zhang, Y. Protein–ligand binding site recognition using complementary binding-specific substructure comparison and sequence profile alignment. *Bioinformatics* **2013**, *29*, 2588–2595.

(12) Yang, J.; Roy, A.; Zhang, Y. BioLiP: a semi-manually curated database for biologically relevant ligand–protein interactions. *Nucleic acids research* **2012**, *41*, D1096–D1103.

(13) Glaser, F.; Pupko, T.; Paz, I.; Bell, R. E.; Bechor-Shental, D.; Martz, E.; Ben-Tal, N. ConSurf: identification of functional regions in proteins by surface-mapping of phylogenetic information. *Bioinformatics* **2003**, *19*, 163–164.

(14) Brylinski, M.; Skolnick, J. A threading-based method (FINDSITE) for ligand-binding site prediction and functional annotation. *Proceedings of the National Academy of sciences* **2008**, *105*, 129–134.

(15) Wass, M. N.; Kelley, L. A.; Sternberg, M. J. 3DLigandSite: predicting ligand-binding sites using similar structures. *Nucleic acids research* **2010**, *38*, W469–W473.

(16) Roche, D. B.; Tetchner, S. J.; McGuffin, L. J. FunFOLD: an improved automated method for the prediction of ligand binding residues using 3D models of proteins. *BMC bioinformatics* **2011**, *12*, 1–20.

(17) Roy, A.; Zhang, Y. Recognizing protein-ligand binding sites by global structural alignment and local geometry refinement. *Structure* **2012**, *20*, 987–997.

(18) Krivák, R.; Hoksza, D. Improving protein-ligand binding site prediction accuracy by classification of inner pocket points using local features. *Journal of cheminformatics* **2015**, *7*, 1–13.

(19) Krivák, R.; Hoksza, D. P2Rank: machine learning based tool for rapid and accurate prediction of ligand binding sites from protein structure. *Journal of cheminformatics* **2018**, *10*, 39.

(20) Capra, J. A.; Laskowski, R. A.; Thornton, J. M.; Singh, M.; Funkhouser, T. A. Predicting protein ligand binding sites by combining evolutionary sequence conservation and 3D structure. *PLoS Computational Biology* **2009**, *5*, e1000585.

(21) Huang, B. MetaPocket: a meta approach to improve protein ligand binding site prediction. *OMICS A Journal of Integrative Biology* **2009**, *13*, 325–330.

(22) Ballester, P. J.; Mitchell, J. B. A machine learning approach to predicting protein–ligand binding affinity with applications to molecular docking. *Bioinformatics* **2010**, *26*, 1169–1175.

(23) Chen, K.; Mizianty, M. J.; Kurgan, L. Prediction and analysis of nucleotide-binding residues using sequence and sequence-derived structural descriptors. *Bioinformatics* **2012**, *28*, 331–341.

(24) Durrant, J. D.; McCammon, J. A. NNScore: a neural-network-based scoring function for the characterization of protein- ligand complexes. *Journal of chemical information and modeling* **2010**, *50*, 1865–1871.

(25) Durrant, J. D.; McCammon, J. A. NNScore 2.0: a neural-network receptor–ligand scoring function. *Journal of chemical information and modeling* **2011**, *51*, 2897–2903.

(26) Chen, P.; Huang, J. Z.; Gao, X. LigandRFs: random forest ensemble to identify ligand-binding residues from sequence information alone. BMC bioinformatics. 2014; pp 1–12.

(27) Wu, Q.; Peng, Z.; Zhang, Y.; Yang, J. COACH-D: improved protein–ligand binding sites prediction with refined ligand-binding poses through molecular docking. *Nucleic acids research* **2018**, *46*, W438–W442.

(28) da Silva, A. D.; Bitencourt-Ferreira, G.; de Azevedo Jr, W. F. Taba: A tool to analyze the binding affinity. *Journal of computational chemistry* **2020**, *41*, 69–73.

(29) Jiménez, J.; Skalic, M.; Martinez-Rosell, G.; De Fabritiis, G. K deep: protein–ligand absolute binding affinity prediction via 3d-convolutional neural networks. *Journal of chemical information and modeling* **2018**, *58*, 287–296.

(30) Öztürk, H.; Özgür, A.; Ozkirimli, E. DeepDTA: deep drug–target binding affinity pre-
diction. *Bioinformatics* **2018**, *34*, i1821–i1829.

(31) Hanson, J.; Paliwal, K.; Litfin, T.; Yang, Y.; Zhou, Y. Accurate prediction of pro-
tein contact maps by coupling residual two-dimensional bidirectional long short-term
memory with convolutional neural networks. *Bioinformatics* **2018**, *34*, 4039–4045.

(32) Wang, S.; Sun, S.; Li, Z.; Zhang, R.; Xu, J. Accurate de novo prediction of protein con-
tact map by ultra-deep learning model. *PLoS computational biology* **2017**, *13*, e1005324.

(33) Senior, A. W.; Evans, R.; Jumper, J.; Kirkpatrick, J.; Sifre, L.; Green, T.; Qin, C.;
Žídek, A.; Nelson, A. W.; Bridgland, A., et al. Improved protein structure prediction
using potentials from deep learning. *Nature* **2020**, *577*, 706–710.

(34) Li, Y.; Zhang, C.; Bell, E. W.; Yu, D.-J.; Zhang, Y. Ensembling multiple raw coevo-
lutionary features with deep residual neural networks for contact-map prediction in
CASP13. *Proteins: Structure, Function, and Bioinformatics* **2019**, *87*, 1082–1091.

(35) Tiwari, A.; Parekh, N. Network-based Machine Learning Approach for Structural Do-
main Identification in Proteins. *bioRxiv* **2020**,

(36) Jiménez, J.; Doerr, S.; Martínez-Rosell, G.; Rose, A. S.; De Fabritiis, G. DeepSite:
protein-binding site predictor using 3D-convolutional neural networks. *Bioinformatics*
2017, *33*, 3036–3042.

(37) Stepniewska-Dziubinska, M. M.; Zielenkiewicz, P.; Siedlecki, P. Improving detection of
protein-ligand binding sites with 3D segmentation. *Scientific reports* **2020**, *10*, 1–9.

(38) Cui, Y.; Dong, Q.; Hong, D.; Wang, X. Predicting protein-ligand binding residues with
deep convolutional neural networks. *BMC bioinformatics* **2019**, *20*, 93.

(39) Aggarwal, R.; Gupta, A.; Chelur, V.; Jawahar, C.; Priyakumar, U. D. DeepPocket:

Ligand Binding Site Detection and Segmentation using 3D Convolutional Neural Networks. **2021**,

(40) He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. Proceedings of the IEEE conference on computer vision and pattern recognition. 2016; pp 770–778.

(41) Desaphy, J.; Bret, G.; Rognan, D.; Kellenberger, E. sc-PDB: a 3D-database of ligandable binding sites—10 years on. *Nucleic acids research* **2015**, *43*, D399–D404.

(42) Berman, H.; Henrick, K.; Nakamura, H. Announcing the worldwide protein data bank. *Nature Structural & Molecular Biology* **2003**, *10*, 980–980.

(43) Needleman, S. B.; Wunsch, C. D. A general method applicable to the search for similarities in the amino acid sequence of two proteins. *Journal of molecular biology* **1970**, *48*, 443–453.

(44) Zhang Lab NW-Align. `http://zhanglab.ccmb.med.umich.edu/NW-align`.

(45) Da Silva, F.; Desaphy, J.; Rognan, D. IChem: A Versatile Toolkit for Detecting, Comparing, and Predicting Protein–Ligand Interactions. *ChemMedChem* **2018**, *13*, 507–510.

(46) Jones, D. T.; Buchan, D. W.; Cozzetto, D.; Pontil, M. PSICOV: precise structural contact prediction using sparse inverse covariance estimation on large multiple sequence alignments. *Bioinformatics* **2012**, *28*, 184–190.

(47) Zhang, C.; Zheng, W.; Mortuza, S.; Li, Y.; Zhang, Y. DeepMSA: constructing deep multiple sequence alignment to improve contact prediction and fold-recognition for distant-homology proteins. *Bioinformatics* **2020**, *36*, 2105–2112.

(48) Mirdita, M.; von den Driesch, L.; Galiez, C.; Martin, M. J.; Söding, J.; Steinegger, M.

Uniclust databases of clustered and deeply annotated protein sequences and alignments. *Nucleic acids research* **2017**, *45*, D170–D176.

(49) Remmert, M.; Biegert, A.; Hauser, A.; Söding, J. HHblits: lightning-fast iterative protein sequence searching by HMM-HMM alignment. *Nature methods* **2012**, *9*, 173–175.

(50) Suzek, B. E.; Wang, Y.; Huang, H.; McGarvey, P. B.; Wu, C. H.; Consortium, U. UniRef clusters: a comprehensive and scalable alternative for improving sequence similarity searches. *Bioinformatics* **2015**, *31*, 926–932.

(51) Johnson, L. S.; Eddy, S. R.; Portugaly, E. Hidden Markov model speed heuristic and iterative HMM search procedure. *BMC bioinformatics* **2010**, *11*, 431.

(52) Potter, S. C.; Luciani, A.; Eddy, S. R.; Park, Y.; Lopez, R.; Finn, R. D. HMMER web server: 2018 update. *Nucleic acids research* **2018**, *46*, W200–W204.

(53) Jones, D. T. Protein secondary structure prediction based on position-specific scoring matrices. *Journal of molecular biology* **1999**, *292*, 195–202.

(54) Jones, D. T.; Singh, T.; Kosciolek, T.; Tetchner, S. MetaPSICOV: combining coevolution methods for accurate prediction of contacts and long range hydrogen bonding in proteins. *Bioinformatics* **2015**, *31*, 999–1006.

(55) Falcon, W. PyTorch Lightning. 2019; `https://github.com/PyTorchLightning/pytorch-lightning`.

(56) Paszke, A.; Gross, S.; Massa, F.; Lerer, A.; Bradbury, J.; Chanan, G.; Killeen, T.; Lin, Z.; Gimelshein, N.; Antiga, L., et al. Pytorch: An imperative style, high-performance deep learning library. *arXiv preprint arXiv:1912.01703* **2019**,

(57) Rego, N.; Koes, D. 3Dmol. js: molecular visualization with WebGL. *Bioinformatics* **2015**, *31*, 1322–1324.

(58) Galicia, C.; Lhospice, S.; Varela, P. F.; Trapani, S.; Zhang, W.; Navaza, J.; Herrou, J.; Mignot, T.; Cherfils, J. MglA functions as a three-state GTPase to control movement reversals of Myxococcus xanthus. *Nature communications* **2019**, *10*, 1–12.

(59) Maldonado, M.; Guo, F.; Letts, J. A. Atomic structures of respiratory complex III2, complex IV, and supercomplex III2-IV from vascular plants. *Elife* **2021**, *10*, e62047.

(60) Czyzyk, D.; Valhondo, M.; Deiana, L.; Tirado-Rives, J.; Jorgensen, W.; Anderson, K. Structure activity relationship towards design of cryptosporidium specific thymidylate synthase inhibitors. *European journal of medicinal chemistry* **2019**, *183*, 111673.

Graphical TOC Entry