Research Article

Application of DNA-Binding Protein Prediction Based on Graph Convolutional Network and Contact Map

Weizhong Lu,[1,2] Nan Zhou,[1] Yijie Ding ⓘ,[1] Hongjie Wu,[1] Yu Zhang,[3] Qiming Fu ⓘ,[1] and Haiou Li[2]

[1]*School of Electronic and Information Engineering, Suzhou University of Science and Technology, Suzhou, China*
[2]*Provincial Key Laboratory for Computer Information Processing Technology, Soochow University, Suzhou, China*
[3]*Suzhou Industrial Park Institute of Services Outsourcing, Suzhou, China*

Correspondence should be addressed to Yijie Ding; wuxi_dyj@163.com

DNA contains the genetic information for the synthesis of proteins and RNA, and it is an indispensable substance in living organisms. DNA-binding proteins are an enzyme, which can bind with DNA to produce complex proteins, and play an important role in the functions of a variety of biological molecules. With the continuous development of deep learning, the introduction of deep learning into DNA-binding proteins for prediction is conducive to improving the speed and accuracy of DNA-binding protein recognition. In this study, the features and structures of proteins were used to obtain their representations through graph convolutional networks. A protein prediction model based on graph convolutional network and contact map was proposed. The method had some advantages by testing various indexes of PDB14189 and PDB2272 on the benchmark dataset.

1. Introduction

The sequence of a protein determines its structure and different structures determine different functions. There is about 18% of the weight of protein in the human body. As the carrier of life, it plays a very important role in human production and life. As a major component of life, proteins are involved in almost all activities of cells, including DNA replication and transcription, chromatin formation, cell growth, and a series of activities, all of which cannot be separated by specific proteins [1]. These proteins that bind to and interact with DNA are called DNA-binding proteins. It has a strong affinity with single-stranded DNA, but a small affinity with double-stranded DNA. Therefore, DNA-binding proteins are also called helical instability proteins, single-stranded DNA-binding proteins [2].

With the development of gene sequencing, various sequencing studies have left many DNA and proteins, including DNA-binding proteins. Using machine learning and deep learning methods to predict DNA-binding proteins has reached a good level, but there is still room for improvement.

At present, many methods based on machine learning have emerged to distinguish DNA-binding proteins, which are divided into structure and sequence methods. Yubo et al. [3] proposed a DBD-Hunter method that combines structural comparison with an assessment of statistical potential to measure the interaction between DNA bases and protein residues. Zhou et al. [4] used random forest for classification by adopting amino acid preservation pattern, potential electrostatic, and other features. However, these methods are too dependent on the protein structure, so the practical operation is difficult. Therefore, sequence-based studies were carried out. Liu et al. [5] proposed a new method for predicting DNA-binding proteins, IDNA-Pro, by integrating features into pseudoamino acids from protein sequences and classifying them through random forest. Zhao et al. [6] classified DNA-binding proteins based on the physicochemical properties of amino acids by using

random forest to recognize the sequence features generated by PseAcc. Although the method based on machine learning can identify DNA-binding proteins well, it needs a lot of human intervention in the process of feature selection and could not properly grasp the relationship between data and features. To overcome this difficulty, deep learning techniques were introduced into protein prediction. Loo et al. [7] proposed a new prediction method MsDBP, which input the fused multiscale features into a deep neural network for learning and classification. The classification was tested with 67% accuracy on a separate dataset PDB2272. Compared with machine learning method, it can save the necessary manual intervention, but the prediction result needs to be improved.

Although there are many methods used to predict DNA-binding proteins at present, the results still have room for improvement. The main problem is how to obtain the high-precision protein structure from the protein sequence, because the accuracy of protein structure and feature has a great impact on the prediction results. In addition, the graph convolution network (GCN) has been widely used in the research of bioinformatics. Graph composed of nodes and edges serves as the input of the network without any requirements on size and format [8]. In order to improve the accuracy of structure and prediction, combining with the current developing trend of the technology of deep learning, a DNA-binding protein prediction model based on GCN and contact map was proposed. The protein graph depends on the sequence of the results of the comparison, so first introducing the preprocess of the dataset, including sequence comparison and filtering; the part of the output is used to calculate the features, and the other part as the input of Pconsc4 model [9], which is used to predict protein contact map, so the inputs of the model are feature matrix and adjacency matrix. We use them for training and prediction. The experimental results show that the prediction performance of DNA-binding proteins can be obtained by the method described. The research content of this paper is shown in Figure 1.

2. Materials and Methods

The prediction of DNA-binding proteins is divided into three parts: data preprocessing, training model, and testing. GCN differs from neural networks in that it introduces a graph structure to represent proteins, which can better represent the structure of proteins. The main purpose of protein sequence preprocessing is to obtain the features and structures of proteins. For the protein processing, the contact map is obtained by predicting the sequence through Pconsc4, and its output exactly corresponds to the adjacency matrix of GCN [10].

2.1. The Dataset. The DNA-binding protein dataset selected is the internationally common dataset. PDB14189 and PDB2272 were established by Gomes et al. [11]. Among them, the PDB14189 dataset was divided into 7129 DNA-binding protein sequences and 7060 DNA-unbinding protein sequences, and the PDB2272 dataset was divided

into 1153 DNA-binding proteins and 1119 nonbinding proteins. PDB14189 was taken as the training set and PDB2272 as the test set. The dataset is detailed in Table 1 below. Among them, positive represents DNA-binding proteins, while negative represents non-DNA-binding proteins.

2.2. Protein Representation. The representation of proteins is generally divided into spatial structure and feature. The long-chain stable structure of protein also contains hydrogen bonds, hydrophobic bonds, salt bonds, and so on [12]. Each protein contains lots of atoms, if each atom is viewed as a node, then the protein graph will be very large, which will increase the pressure of training and is not easy to achieve. However, there are about hundreds of residues in a protein, and there is no other spatial information between residues, so it is more suitable to be used as nodes to represent structural features. The spatial structure of a protein can be represented by a contact map; it represents the two-dimensional structure of the protein; each element in the matrix represents the probability of contact at the corresponding position [13]; the value is between 0 and 1. Figure 2 shows a protein contact map.

Predicting the structure of a protein from its sequence is the purpose of introducing contact map. Specifically, assuming that the length of protein sequence is M, the size of its contact map is $M * M$. $M(i, j)$ represents the probability of contact between the ith residue and the jth residue. If the value is less than the threshold value, it can be considered that they are in contact. Pconsc4 is a fast and efficient method to predict contact map. Since its output is a probability value between 0 and 1, the threshold value of 0.5 was set for the obtained contact maps, and the probability value greater than or equal to 0.5 was set as 1.The rest were set as 0, so that the structural information of the protein could be well extracted, corresponding to the adjacency matrix as the input GCN network [14].

The next step is the extraction of protein features. Since residues are used as nodes, the properties of residues are selected as features. Due to the differences in the R group, different features are displayed, including aromaticity, polarity, and explicit valence [15]. Position-specific scoring matrix (PSSM) is a commonly used representation of protein features, in which the results of each element depend on the results of sequence comparison, and these results represent the feature of proteins [16]. Other features were also used, such as the primary thermal coding of the remaining symbols, whether the residue was aromatic, whether the residue was acidic charged, and whether it was extremely neutral, etc. [17], as shown in Table 2. In summary, the total number of features is 54, so the protein's feature matrix dimension is $(M, 54)$.

For PSSM, the basic position frequency matrix (PFM) [18] is calculated by the number of occurrence of residues at each position in the sequence of sequence alignment results. Equation (1) is as follows:

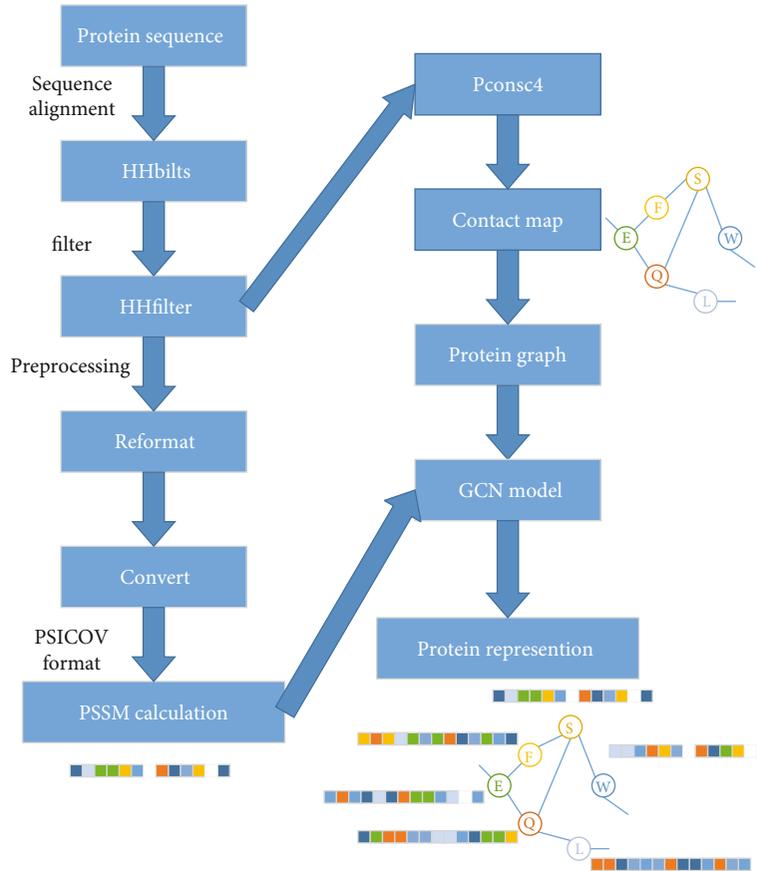$$M_{k,j}^{\text{PFM}} = \sum_{i=1}^{N} I\left(A_{i,j} = k\right), \tag{1}$$

FIGURE 1: The processing of proteins, including the preprocessing of sequence, the generation of graph structures, and feature extraction, Pconsc4 was used to extract protein structural information. Finally, protein graph was generated higher-level feature graph through GCN.

TABLE 1: Introduction to the dataset.

Number\dataset	PDB14189	PDB2272
Positive	7129	1153
Negative	7060	1119
Total	14189	2272

FIGURE 2: The contact map of protein.

where A represents a set of alignment sequences equal to the target protein length, k is the set of residues, $i = (1, 2 \cdots, N)$, $j = (1, 2, \cdots L)$, and $i(x)$ is the indicator function when the condition is met or not. Equation (2) is used to obtain the position probability matrix (PPM):

$$M_{k,j}^{\text{PPM}} = \frac{M_{k,j}^{\text{PFM}} + (p/4)}{N + p}. \tag{2}$$

In order to prevent the matrix entries from appearing 0, according to human experience, the pseudocount [19] p was set 0.8, so that PPM was regarded as a part of the node features.
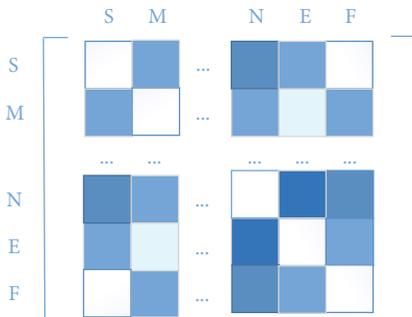
2.3. Model Architecture. Although traditional convolution techniques perform well for Euclidean data, they perform poorly for non-Euclidean data [20]. Therefore, graph convolution technology came into being. For a graph, the edges of each node are related to other nodes and this information can be used to capture interdependencies between instances, so the node can aggregate its own features and its neighbor features to generate a new representation of the node [21]. With the continuous development of graph learning, there are many variations, like GAT, GAE, and GGN [22]. All these network models can extract the feature; for using the GCN layer, each layer convolution operation is as shown in Equation (3):

$$H^{l+1} = f\left(H^l, A\right) = \sigma\left(D\wedge^{-1/2}\hat{A}D\wedge^{-1/2}H^lW^{l+1}\right). \tag{3}$$

Among them, A is the adjacency matrix of node features,

TABLE 2: Node features.

| Label	Feature	Size
1	One-hot encoding of the residue symbol	21
2	Position-specific scoring matrix (PSSM)	21
3	Whether the residue is aliphatic	1
4	Whether the residue is aromatic	1
5	Whether the residue is polar neutral	1
6	Whether the residue is acidic charged	1
7	Whether the residue is basic charged	1
8	Residue weight	1
9	The negative of the logarithm of the dissociation constant for the –COOH group	1
10	The negative of the logarithm of the dissociation constant for the –NH3 group	1
11	The negative of the logarithm of the dissociation constant for any other group in the molecule	1
12	The pH at the isoelectric point	1
13	Hydrophobicity of residue (pH = 2)	1
14	Hydrophobicity of residue (pH = 7)	1
	Total	54
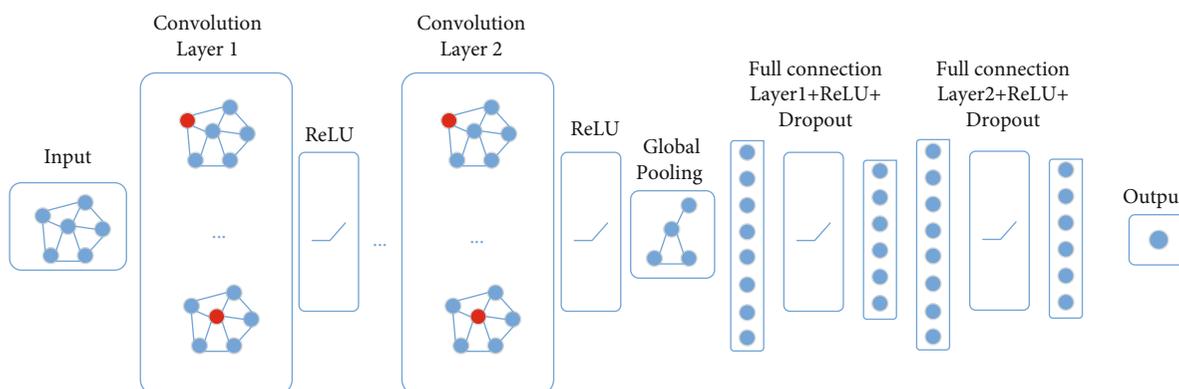
FIGURE 3: The structure of the GCN network, graphs of DNA-binding proteins through the GCN to get their representation.

assuming that the node number is m, then its adjacency matrix is (m, m), $\widehat{D}$ is the degree of matrix (m, m), which represents the connection relationship between residues, $\widehat{D} = D + I$, I is a unit matrix, considers itself features, W^{l+1} is the first $l + 1$ layer of weighting matrix, H^l is the output of the first layer of l, and $H^0 = X$, X is the input of the feature matrix, Figure 3 shows the architecture of the model.

The protein graph contained much information about the interactions and positions of each residue pair, which was important for feature learning and predicting DNA-binding proteins. It was input into the GCN to extract the features. After convolution of multiple GCN layers, the representation of protein was effectively extracted. Then, the overall features of protein for prediction were obtained. The prediction includes two full connection layers. The results were presented as probabilities.

Using GCN to map proteins to the representation of rich features has also become a method of protein feature extraction. In addition, there were many factors affecting the experimental results, such as dropout, epoch, and batch.

TABLE 3: The hyperparameter settings using human experience.

Hyperparameter	Setting
Epoch	1000
Batch size	128
Learning rate	0.001
Optimizer	Adam
The number of convolution layers	3
Fully connected layers after GCN	2

TABLE 4: Combinations of GCN models on PDB14189.

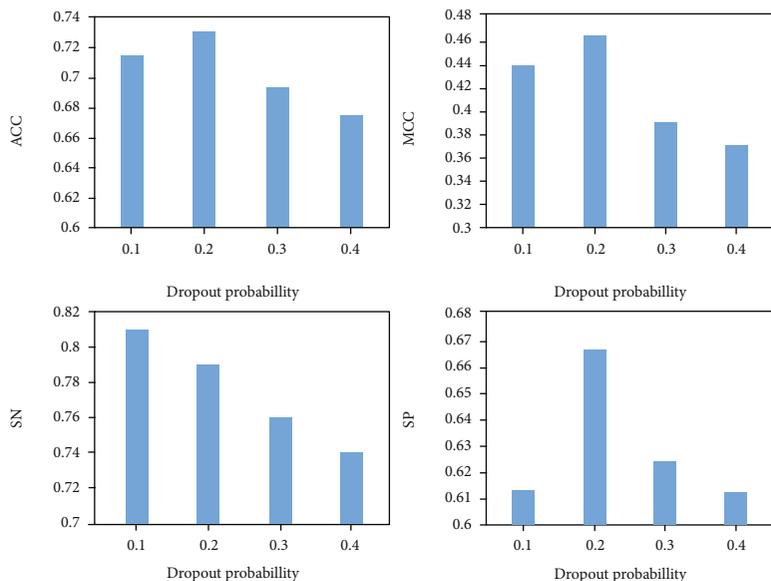| Model	Number of layers	Layer1 (in, out)	Layer2 (in, out)	Layer3 (in, out)
GCN	1	(54,54)	—	—
GCN	2	(54,54)	(54,108)	—
GCN	3	(54,54)	(54,108)	(108,216)

FIGURE 4: Comparison of prediction performance of different dropout probabilities.

The setting of some hyperparameters were compared and determined through experiments.

3. Results and Discussion

The experiment was built on PyTorch [23], an open source deep learning framework. The GCN model was based on its PyG implementation [24], PDB14189 was used for testing to find the optimal super parameters, and PDB2272 was used to test model performance.

3.1. The Evaluation Index. Accuracy (ACC), Matthews correlation coefficient (MCC), sensitivity (SN), and specificity (SP) were used as the evaluation indexes of the model [25], these indexes were widely used in the studies of biological sequences, as shown in

$$
\begin{cases}
SN = \dfrac{TP}{TP + FN}, \\[2mm]
SP = \dfrac{TN}{TN + FP}, \\[2mm]
ACC = \dfrac{TP + TN}{TP + FP + TN + FN}, \\[2mm]
MCC = \dfrac{TP \times TN - FP \times FN}{\sqrt{(TP + FN) \times (TN + FP) \times (TP + FP) \times (TN + FN)}}.
\end{cases}
\tag{4}
$$

Among them, TP is the number of the correctly predicted positive samples, TN is the number of the correctly predicted negative samples, FP is the number of the wrongly predicted positive samples, and FN is the number of the wrongly predicted negative samples. SN represents the percentage of correctly predicted positive samples, SP represents the percentage of correctly predicted negative samples, ACC represents the percentage of correctly predicted samples in total samples, and MCC represents the prediction quality of the binary classification model, with a range of [−1, 1]. The larger the MCC is, the better the prediction quality of the model is.

3.2. The Setting of Hyperparameters. Training an optimal model requires constantly adjusting the hyperparameters of the model, which can be modified based on human experience. Some of the hyperparameters were shown in Table 3. In this model, according to human experience, the GCN layer was set to three, dimensions of input and output for each layer were shown in Table 4. Some other parameters were compared in the following experiences.

3.3. Model Performance when Selecting Different Dropouts. After protein feature extraction, in order to better improve the accuracy of classification, two full connection layers were added to the ends to improve the learning ability of the model. In the fully connected layer, in order to avoid overfitting of the model, dropout was introduced to shut down some neurons with a probability value. Different probability values will affect the performance of prediction. To evaluate the impact of different dropout values, Figure 4 shows the performance of the model according to different dropout values. When the dropout is 0.2, the model has the highest performance compared to other parameters.

3.4. Whether PSSM Is Included in Feature Selection. The selection of protein feature greatly affects the accuracy of prediction. Since the dimension of PSSM matrix constructed by features was very small, the experiment was carried out with PSSM or without PSSM. Figure 4 shows the results of various indicators under the condition. PSSM depends on the sequence correlation results, which contains much evolutionary information about the sequence, and ultimately determines the protein features. As can be seen from Figure 5, PSSM can effectively represent the features of proteins and effectively improve the prediction performance.
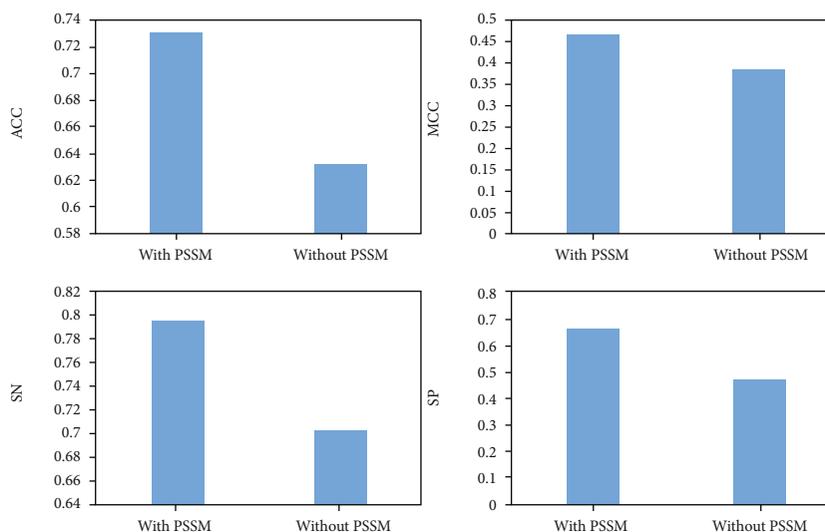
Figure 5: Comparison of performance results with or without PSSM.

Table 5: Comparison between the proposed method and existing methods on PDB2272.

Methods	ACC (%)	MCC (%)	SN (%)	SP (%)
Qu et al. [26]	48.33	3.34	48.31	48.35
Local-DPP [27]	50.57	4.56	8.76	93.66
Pse-DNA-Pro [28]	61.88	24.30	75.28	48.08
DPP-Pse-AAC [29]	58.10	16.25	56.63	59.61
Ms-DBP [30]	66.99	33.97	70.69	63.18
GCN-method	78.49	59.27	92.59	64.15

3.5. Analysis of Experimental Results. In the independent test dataset, PDB14189 was used as the training dataset to train the model, and PDB2272 was used as the test dataset. According to the optimal experimental parameters, the final DNA-binding protein classification model was constructed: the number of GCN layers were three, dropout was 0.2, PSSM was selected as the feature, the input and output dimensions of each layer were $(54, 54)$, $(54, 108)$, and $(108, 216)$. Other methods were compared with the method, and the method reached ACC (78.49%), SN (92.59%), SP (64.15%), and MCC (59.27%). Under certain conditions, the method has certain advantages compared with the existing methods, as shown in Table 5.

4. Conclusions

DNA-binding proteins are enzymes, which can bind with DNA to produce complex proteins and play important roles in the functions of a variety of biological molecules. In order to improve the accuracy of prediction of DNA-binding protein, a DNA-binding protein prediction model based on GCN and contact map was proposed. In this model, the dataset was preprocessed by sequence alignment; then, the structural information is extracted by Pconsc4 model; PSSM and some biological characteristics are used as features. Finally, the GCN model was constructed to train and predict DNA-binding protein data. The protein graph contained information about the interactions and positions of each residue pair, which was important for feature learning and predicting binding proteins. The protein graph was input into the GCN to extract the features, and the prediction included two full connection layers. Using GCN to map proteins to the representation of rich features has also become a method of protein feature extraction. Through training and parameter tuning, the performance of GCN model was better than some existing methods. It also provides some thoughts for other fields of biological information.

In the future, we plan to carry out a research on feature extraction and network model to improve the accuracy of DNA-binding proteins and related prediction. Different biological features can be combined, and methods such as attention mechanism can be considered to improve the model, in order to achieve the goal of improving the prediction effect and other indicators.

Data Availability

The datasets can be found in the references.

Conflicts of Interest

The authors declare that there is no conflict of interest regarding the publication of this paper.

Acknowledgments

References

[1] A. S. Rifaioglu, H. Atas, M. J. Martin, R. Cetin-Atalay, V. Atalay, and T. Doğan, "Recent applications of deep learning and machine intelligence on in silico drug discovery: methods, tools and databases," *Briefings in Bioinformatics*, vol. 20, no. 5, pp. 1878–1912, 2019.

[2] M. S. Nogueira and O. Koch, "The development of target-specific machine learning models as scoring functions for docking-based target prediction," *Journal of Chemical Information and Modeling*, 2019.

[3] Y. Wang, Y. Ding, F. Guo, L. Wei, and J. Tang, "Improved detection of DNA-binding proteins via compression technology on PSSM information," *PLoS One*, vol. 12, no. 9, 2017.

[4] L. Zhou, X. Song, D. J. Yu, and J. Sun, "Sequence-based detection of DNA-binding proteins using Multiple-view features allied with feature selection," *Molecular Informatics*, vol. 39, no. 8, p. 2000006, 2020.

[5] K. Liu, X. Sun, L. Jia et al., "Chemi-net: a Molecular graph convolutional network for accurate drug property prediction," *International Journal of Molecular Sciences*, vol. 20, no. 14, p. 3389, 2019.

[6] H. Zhang, Q. Zhang, F. Ju et al., "Correction to: Predicting protein inter-residue contacts using composite likelihood maximization and deep learning," *BMC Bioinformatics*, vol. 20, no. 1, p. 616, 2019.

[7] J. Loo, A. L. Emtage, L. Murali, S. S. Lee, A. L. W. Kueh, and S. P. H. Alexander, "Ligand discrimination during virtual screening of the CB1 cannabinoid receptor crystal structures following cross-docking and microsecond molecular dynamics simulations," *RSC Advances*, vol. 9, no. 28, pp. 15949–15956.

[8] M. Michel, D. Menéndez Hurtado, and A. Elofsson, "PconsC4: fast, accurate, and hassle-free contact predictions," *Bioinformatics (Oxford, England)*, vol. 35, no. 15, pp. 2677–2679, 2019.

[9] L. Jiang, S. Wang, B. Zhang et al., ""A more probable explanation" is still impossible to explain GN-z11-flash: in response to Steinhardt et al. (arXiv:2101.12738)," 2021, https://arxiv.org/abs/2102.01239.

[10] S. Wang, S. Sun, Z. Li, R. Zhang, and J. Xu, "Accurate de novo prediction of protein contact map by ultra-deep learning model," *PLoS Computational Biology*, vol. 13, no. 1, article e1005324, 2017.

[11] J. Gomes, B. Ramsundar, E. N. Feinberg, and V. S. Pande, "Atomic convolutional networks for predicting protein-ligand binding affinity," https://arxiv.org/abs/1703.10603, 2017.

[12] R. Gómez-Bombarelli, J. N. Wei, D. Duvenaud et al., "Automatic chemical design using a data-driven continuous representation of molecules," *ACS Central Science*, vol. 4, no. 2, pp. 268–276, 2018.

[13] E. B. Lenselink, N. Ten Dijke, B. Bongers et al., "Beyond the hype: deep neural networks outperform established methods using a ChEMBL bioactivity benchmark set," *Journal of Cheminformatics*, vol. 9, no. 1, p. 45, 2017.

[14] V. Le, T. P. Quinn, T. Tran, and S. Venkatesh, "Deep in the bowel: highly interpretable neural encoder-decoder networks predict gut metabolites from gut microbiome," *BMC Genomics*, vol. 21, no. S4, 2020.

[15] Z. Hakime, Z. Arzucan, and O. Elif, "DeepDTA: deep drug-target binding affinity prediction," *Bioinformatics*, vol. 17, p. 17, 2018.

[16] M. Sun, S. Zhao, C. Gilvary, O. Elemento, J. Zhou, and F. Wang, "Graph convolutional networks for computational drug development and discovery," *Briefings in Bioinformatics*, vol. 21, no. 3, pp. 919–935, 2020.

[17] T. Wen and R. B. Altman, "Graph convolutional neural networks for predicting drug-target interactions," *Journal of Chemical Information and Modeling*, vol. 59, no. 10, pp. 4131–4149, 2019.

[18] T. Nguyen, H. Le, and S. Venkatesh, "GraphDTA: prediction of drug-target binding affinity using graph convolutional networks," *BioRxiv*, vol. 2019, p. 684662, 2019.

[19] K. Nishida, M. C. Frith, and K. Nakai, "Pseudocounts for transcription factor binding sites," *Nucleic Acids Research*, vol. 37, no. 3, pp. 939–944, 2009.

[20] K. Xu, W. Hu, J. Leskovec, and S. Jegelka, "How powerful are graph neural networks?," 2018, https://arxiv.org/abs/1810.00826.

[21] C. Shen, Y. Ding, J. Tang, J. Song, and F. Guo, "Identification of DNA–protein binding sites through multi-scale local average blocks on sequence information," *Molecules*, vol. 22, no. 12, p. 2079, 2017.

[22] J. Hanson, T. Litfin, K. Paliwal, and Y. Zhou, "Identifying molecular recognition features in intrinsically disordered regions of proteins by transfer learning," *Bioinformatics*, vol. 36, no. 4, 2019.

[23] A. Paszke, S. Gross, S. Chintala et al., *Automatic differentiation in PyTorch*, 2017.

[24] S. Akbar, S. Khan, F. Ali, M. Hayat, M. Qasim, and S. Gul, "iHBP-DeepPSSM: identifying hormone binding proteins using PsePSSM based evolutionary features and deep learning approach," *Chemometrics and Intelligent Laboratory Systems*, vol. 204, article 104103, 2020.

[25] T. Song, S. Wang, D. Liu et al., "SE-OnionNet: a convolution neural network for protein–ligand binding affinity prediction," *Frontiers in Genetics*, vol. 11, article 607824, 2021.

[26] Y. Qu, J. A. Fitzgerald, H. Rauter, and N. Farrell, "Approaches to selective DNA binding in polyfunctional dinuclear platinum chemistry. The synthesis of a trifunctional compound and its interaction with the mononucleotide 5'-guanosine monophosphate," *Inorganic Chemistry*, vol. 40, no. 24, pp. 6324–6327, 2001.

[27] L. Wei, J. Tang, and Q. Zou, "Local-DPP: an improved DNA-binding protein prediction method by exploring local evolutionary information," *Information Sciences*, vol. 384, pp. 135–144, 2017.

[28] B. Liu, J. Xu, S. Fan, R. Xu, J. Zhou, and X. Wang, "PseDNA-pro: DNA-binding protein identification by combining Chou's PseAAC and physicochemical distance transformation," *Molecular Informatics*, vol. 34, no. 1, pp. 8–17, 2015.

[29] Y. D. Khan, M. Jamil, W. Hussain, N. Rasool, S. A. Khan, and K. C. Chou, "pSSbond-PseAAC: prediction of disulfide bonding sites by integration of PseAAC and statistical moments," *Journal of Theoretical Biology*, vol. 463, pp. 47–55, 2019.

[30] X. du, Y. Diao, H. Liu, and S. Li, "MsDBP: exploring DNA-binding proteins by integrating Multiscale sequence information via Chou's Five-Step rule," *Journal of Proteome Research*, vol. 18, no. 8, pp. 3119–3132, 2019.