bioRxiv preprint doi: https://doi.org/10.1101/2021.07.10.451856; this version posted July 12, 2021. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under a CC-BY-NC-ND 4.0 International license.

picture(0,0)(-35,0)(1,0)30 (0,35)(0,-1)30 picture picture(0,0)(35,0)(-1,0)30 (0,35)(0,-1)30 picture

Bioinformatics
doi.10.1093/bioinformatics/xxxxxx
Advance Access Publication Date: Day Month Year
Manuscript Category

OXFORD

Sequence analysis

Attention-based convolutional neural networks for protein-protein interaction site prediction

Shuai Lu [1], Yuguang Li [2], Xiaofei Nan [1,*], and Shoutao Zhang [3,*]

[1] School of Information Engineering, Zhengzhou University, Zhengzhou, 45001, China and
[2] College of Economics and Management, Zhengzhou University of Light Industry, Zhengzhou, 45001, China and
[3] School of Life Science, Zhengzhou University, Zhengzhou, 45001, China.

* To whom correspondence should be addressed.

Associate Editor: XXXXXXX

Received on XXXXX; revised on XXXXX; accepted on XXXXX

Abstract

Motivation: Protein-protein interactions are of great importance in the life cycles of living cells. Accurate prediction of the protein-protein interaction site (PPIs) from protein sequence improves our understanding of protein-protein interaction, contributes to the protein-protein docking and is crucial for drug design. However, practical experimental methods are costly and time-consuming so that many sequence-based computational methods have been developed. Most of those methods employ a sliding window approach, which utilize local neighbor information within a window size. However, they don't distinguish and use the effect of each individual neighboring residue at different position.
Results: We propose a novel sequence-based deep learning method consisting of convolutional neural networks (CNNs) and attention mechanism to improve the performance of PPIs prediction. Our attention-based CNNs captures the different effect of each neighboring residue within a sliding window, and therefore making a better understanding of the local environment of target residue. We employ experiments on several public benchmark datasets. The experimental results demonstrate that our proposed method significantly outperforms the state-of-the-art techniques. We also analyze the difference using various sliding window sizes and amino acid residue features combination.
Availability: The source code can be obtained from https://github.com/biolushuai/attention-based-CNNs-for-PPIs-prediction
Contact: iexfnan@zzu.edu.cn or zhangst@zzu.edu.cn
Supplementary information: Supplementary data are available online.

1 Introduction

Proteins carry out their cellular functions by interacting with a variety of other molecules including DNA, RNA, small ligands and other proteins(Chen and Kurgan, 2009; Ding et al., 2010; Sudha et al., 2014; Fornes et al., 2014). Protein–protein interactions drive many life processes, such as immune response(Mian et al., 1991), transport(Rohani and Zydney, 2009), and signal transduction(Kyriakis and Avruch, 2001). Protein–protein interactions are driven by forming chemical bonds. The bonding amino acid residues participating in interactions are protein-protein interaction sites(Li and Ilie, 2020). Accurate recognition of these interaction site is helpful for the development of novel therapeutics(Kozma et al., 2013; Petta et al., 2016), annotation of protein functions(Orii and Ganapathiraju, 2012), and molecular mechanisms study of diseases(Nibbe et al., 2011; Kuzmanov and Emili, 2013).

However, practical experimental methods for judging whether a residue of protein sequence belongs to the binding sites remains costly and time-consuming(Terentiev et al., 2009; Brettner and Masel, 2012; Wodak et al., 2013). Therefore, a lot of computational predictors for protein-protein interaction site (PPIs) prediction have been developed to bridge the gap. These methods can be roughly divided into three groups: protein-protein docking, structure-based interface prediction and sequence-based interaction sites determination(Hou et al., 2017; Zeng et al., 2020). While protein-protein docking and structure-based methods usually need the 3D structure of proteins, sequence-based approaches are faster and cheaper as they only utilize information from protein sequence. Also, compared with

© The Author 2015.

bioRxiv preprint doi: https://doi.org/10.1101/2021.07.10.451856; this version posted July 12, 2021. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under a CC-BY-NC-ND 4.0 International license.

picture(0,0)(-35,0)(1,0)30 (0,35)(0,-1)30 picture picture(0,0)(35,0)(-1,0)30 (0,35)(0,-1)30 picture

structure information, there are massive protein sequences because of the popularity of high-throughput sequencing techniques.

In this study, we focus on the sequence-based computational models for PPIs prediction, most of which apply machine learning algorithms, including shallow neural networks(Ofran and Rost, 2003, 2007; Sanchez, 2006; Singh G, Dhole K, Pai PP, 2014), random forest(Wei et al., 2015, 2016; Hou et al., 2017), Naïve Bayes(Murakami and Mizuguchi, 2010), logistic regression(Dhole et al., 2014) and ensemble learning(Wei et al., 2016; Zhang and Kurgan, 2019; Li and Ilie, 2020; Zhang et al., 2020). A lot of models in those studies employ a sliding window to gather information from the local environment of target residue. However, the sliding window approaches consider all the neighboring residues having the same effect for PPIs prediction. In fact, each neighboring residue at different position plays a different role in PPIs prediction. In recent years, a new direction of neural network research has emerged. It learns to put different "attention" on different parts of input, and has been applied in machine translation(Bahdanau et al., 2015), sentence classification(Zhao and Wu, 2016), compound–protein interaction prediction(Chen et al., 2020), paratope prediction(Pittala and Bailey-Kellogg, 2020) and protein structure prediction(Berkeley et al., 2020).

Here, we propose a novel deep learning model using attention based convolutional neural networks(CNNs) for PPIs Prediction, where attention mechanism is used to capture the neighboring residue information and its correlation with the target residue. The experiment is carried out on benchmark datasets, and our model outperforms the state-of-the-art methods. In order to investigate whether the attention mechanism is helpful to predict PPI sites, we remove it in our model for comparison purpose. The detailed analyses show that attention mechanism is very important in our model.

2 Materials and Methods

2.1 Datasets

In this study, we use the same protein sequences as the state-of-the-art method DeepPPISP(Zeng et al., 2020), including three benchmark datasets Dset_186, Dset_72(Murakami and Mizuguchi, 2010) and PDBset_164(Singh G, Dhole K, Pai PP, 2014). All the protein sequences in those datasets are collected form Protein Data Bank(Sussman et al., 1999). The sequences are derived from heterodimeric structures with resolution <3.0 Å, sequence homology <25% and length >50. Protein structures with missing ratio (i.e. the number of missing residues of a chain listed in REMARK465 divided by the total number of residues of the chain) >30% are removed. Also, protein complexes with interface area of $<500Å^2$ or $\geq 2500Å^2$ as mentioned in PDBsum(Laskowski, 2009) and transmembrane proteins listed in PDBTM(Kozma et al., 2013) are removed. Same as the works in (Murakami and Mizuguchi, 2010; Singh G, Dhole K, Pai PP, 2014; Zeng et al., 2020), an amino acid residue is defined as an interaction site if its absolute solvent accessibility is $<1Å^2$, before and after the binding form; otherwise, it is defined as a non-interaction site.

Table 1. Sumarry of datasets

Datasets	NO. of Protein Sequences	NO. of Binding Residues	NO. of non-Binding Residues
TrainingSet	352	11206	61982
TestingSet	70	2330	9461

These three datasets come from different resources, and are integrated to a fused dataset to insure that training set and testing set are from an identical distribution, same as DeepPPISP(Zeng et al., 2020). The reconstruction of datasets is helpful for making full use of all proteins to train a deep learning model. Table 1 shows the size of datasets, the number of binding residues and non-binding residues.

2.2 Input Features

In this work, we use residue features including one-hot encoding of protein sequence, protein evolutionary information, and other predicted structural features from protein primary sequence. Because we focus only on the sequence-based computational method, the secondary structure information used here are predicted by NetsurfP-2.0(Klausen et al., 2019) rather than assigned by DSSP(Kabsch and Sander, 1983) which takes protein structure as input. We also take advantage of other features generated by NetsurfP-2.0(Klausen et al., 2019), such as solvent accessibility and backbone dihedral angles. All those features are described in detail as follows:

2.2.1 One-hot encoding of protein sequence

There are only 20 possible natural residue types, and we encode each raw protein sequence as a 20 dimensional one-hot vector, where each element is either 1 or 0 and 1 indicates the existence of a corresponding residue.

2.2.2 Position-specific scoring matrix

Various related works have proved that the evolutionary information in position-specific scoring matrix (PSSM) is helpful for PPIs prediction(Zhang and Kurgan, 2019; Hou et al., 2017; Zeng et al., 2020). By running PSI-BLAST(Altschul et al., 1997) against the non-redundant(McGinnis and Madden, 2004) database with three iterations and an E-value threshold of 0.001, we get PSSM in which each residue is encoded as a 20D vector representing the probabilities of 20 natural residues occurring at each position. For each protein sequence with L residues, there are L rows and 20 columns in PSSM.

2.2.3 Predicted structural features

In this study, we utilize NetsurfP-2.0(Klausen et al., 2019) to predict local structural features from protein primary sequence, including solvent accessibility, secondary structure, and backbone dihedral angles for each residue of the input sequences. NetsurfP-2.0(Klausen et al., 2019) is a novel deep learning model trained on several independent datasets and achieves the state-of-the-art performance of predicting those local structural protein features.

For every reside in each input sequence, we calculate its absolute and relative solvent accessibility surface accessibility(ASA and RSA, respectively) , 8-class secondary structure classification (SS8), and the backbone dihedral angles(ϕ and ψ). ASA and RSA represent the solvent accessibility of an amino acid residue. The predicted secondary structure describes the local structural environment of a residue. And, ϕ and ψ figure the relative positions of adjacent residues. The 8-class secondary structures are: 3-helix (G), a-helix (H), p-helix (I), b-strand (E), b-bridge (B), b-turn (T), bend (S) and loop or irregular (L). All those residues features are very helpful for PPIs prediction.

In our work, a 52D feature vector for each residue in every input protein sequence is used. These features are richer than those in the state-of-the-art method DeepPPISP(Zeng et al., 2020).

2.3 Model architecture

The PPIs prediction problem can be summarized as a binary classification task: judging whether a residue from a given protein sequence binding with its partner protein or not. As described in Section 2.2, each residue is encoded into a 52D vector. And each protein sequence can be represented as a matrix S, including a list of residues:

picture(0,0)(-35,0)(1,0)30 (0,-35)(0,1)30 picture picture(0,0)(35,0)(-1,0)30 (0,-35)(0,1)30 picture

bioRxiv preprint doi: https://doi.org/10.1101/2021.07.10.451856; this version posted July 12, 2021. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under a CC-BY-NC-ND 4.0 International license.

picture(0,0)(-35,0)(1,0)30 (0,35)(0,-1)30 picture picture(0,0)(35,0)(-1,0)30 (0,35)(0,-1)30 picture

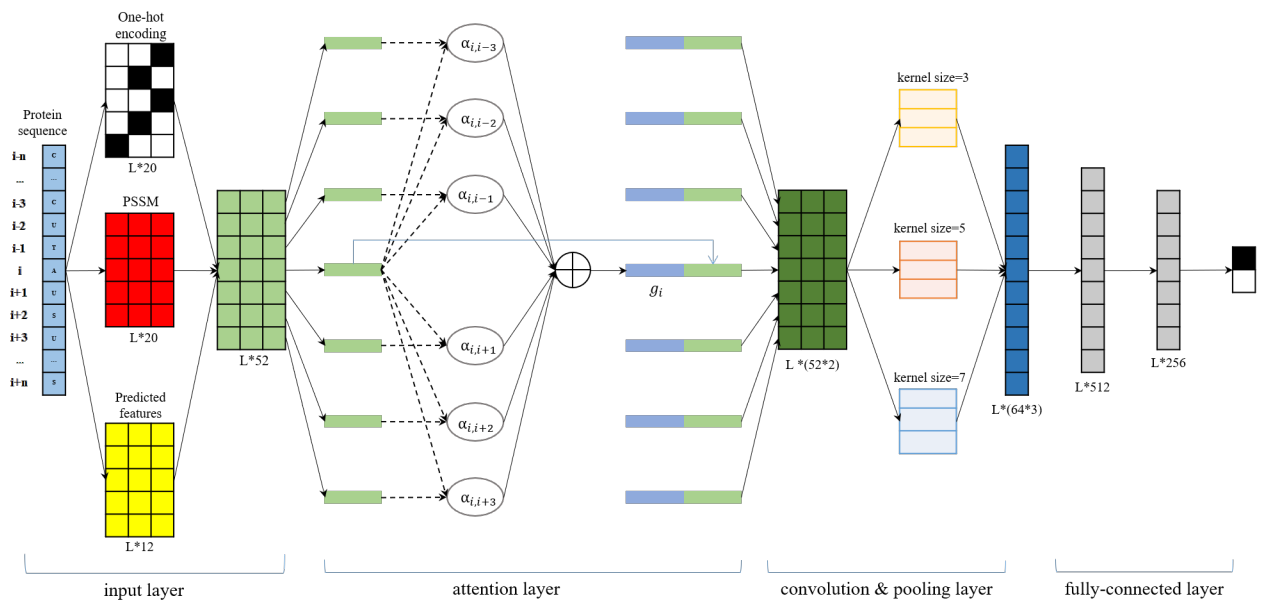

Fig. 1. The deep neural network architecture of our model. In attention layer, the dotted lines correspond to formula (4)-(5), the black solid lines to formula (3), the blue solid line is copy operation.

$$S = [r_1, r_2, r_3, \cdots, r_i, \cdots, r_l]^T, S \in R^{(l*d)} \tag{1}$$

where $r_i \in R^d$ is the residue feature vector corresponding to the i-th residue in the protein sequence, l is the protein sequence length, and d is the residue feature dimension (52 in this paper).

2.3.1 Convolution neural networks

Convolution neural networks(CNNs) model has been adapted to various bioinformatics tasks such as protein binding site prediction(Wardah et al., 2020), protein-ligand scoring(Ragoza et al., 2017) and protein-compound affinity prediction(Karimi et al., 2019). In PPIs prediction, the input of CNNs is a protein sequence represented as a matrix S. The CNNs model employs a convolutional operation on a sliding window of length $w(w = 2n + 1)$, where n could be any positive integer for the i-th target residue r_i. It means that we consider the target amino acid at the center and $2n$ neighboring residues as input features of the target residue.Similar as DeepPPISP(Zeng et al., 2020), we use the all-zero vector padding for those amino acids which do not have neighboring residues in the left or right. A convolutional operation can be shown as:

$$c_i = f(w * r_{i-n:i+n} + b) \tag{2}$$

where f is a non-linear activation function, and $r_{i-n:i+n}$ denotes the concatenation of w residue vectors:$r_{i-n:i+n} = r_{i-n} \oplus r_{i-n+1} \oplus \cdots \oplus r_i \oplus \cdots \oplus r_{i+n-1} \oplus r_{i+n}$, where r_i is the target residue, and $r_{i-n}, r_{i-n+1}, \cdots, r_{i-1}, r_{i+1}, \cdots, r_{i+n}$ are neighboring residues. After convolutional operation, a max pooling operation is applied on c_i, and two fully connected layers are followed which predicts the interaction probability of the target residue. However, CNNs model treats neighboring residues within a sliding window equally by direct concatenation, and ignores their different effects on the target residue.

2.3.2 Attention-based Convolution neural networks

As shown in Fig.1, our model consists of four parts, i.e. input, attention, convolution & polling and fully-connected layers. The input layer takes the protein sequence as input and extract all sequence-based features. The convolution & polling and fully-connected layers share the same architecture as CNNs model said in Section 2.3.1. The attention layer utilizes attention mechanism which is employed as an additional fully-connected layer, and it is trained in conjunction with all the other network components like other machine learning works (Zhao and Wu, 2016). To be more specific, the attention layer is to build a context vector representation for each residue. The context vector is concatenated with the input residue vector to form a new residue representation, and it is then fed to the convolution & polling layer.

The attention mechanism enables a target residue learning to pay different attention on each neighboring residue. Neighboring residue vectors are scored and combined in a weighted sum:

$$g_i = \sum_{j \neq i} \alpha_{i,j} * r_j \tag{3}$$

where $\alpha_{i,j}$ are attention weights and should satisfy the following restrictions: $\alpha_{i,j} \geq 0$ and $\sum_i \alpha_{i,j} = 1$. Context vector g_i is derived for target residue r_i. g_i is shown as a blue rectangle in Fig. 1.

The attention weights are calculated as:

$$\alpha_{i,j} = \frac{exp(score(r_i, r_j))}{\sum_{j'}(score(r_i, r_j)} \tag{4}$$

$$score(r_i, r_j) = v_a^T \tanh(W_a[r_i \oplus r_j]) \tag{5}$$

The residue pair correlation score $score(r_i, r_j)$ is computed by a two-layer neural network described in formula (5). And neighboring residues $r_{j,j \neq i}$ with larger scores contribute more on context vector g_i.

After attention layer, each residue r_i is concatenated with its context vector g_i to form an extended residue vector $r'_i : r'_i = r_i \oplus g_i$, r'_i is then used to fed into the convolution & polling layer. Attention mechanism determines which residues should be put more attention on than other residues over the sequence in PPIs prediction. At last, the fully-connected layer output the binding probability of each target residue.

picture(0,0)(-35,0)(1,0)30 (0,-35)(0,1)30 picture picture(0,0)(35,0)(-1,0)30 (0,-35)(0,1)30 picture

bioRxiv preprint doi: https://doi.org/10.1101/2021.07.10.451856; this version posted July 12, 2021. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under a CC-BY-NC-ND 4.0 International license.

picture(0,0)(-35,0)(1,0)30 (0,35)(0,-1)30 picture picture(0,0)(35,0)(-1,0)30 (0,35)(0,-1)30 picture

2.4 Performance assessment

We use six evaluation metrics to evaluate the performances of the PPIs prediction models. Five of them are classic threshold-dependent binary classification metrics as shown in the following:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{6}$$

$$Precision = \frac{TP}{TP + FP} \tag{7}$$

$$Recall = \frac{TP}{TP + FN} \tag{8}$$

$$F - measure = \frac{2 * Precision * Recall}{Precision + Recall} \tag{9}$$

$$MCC = \frac{TP * TN - FP * FN}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}} \tag{10}$$

where, TP(True Positive) is the number of interacting residues that are correctly predicted as interacting, FP(False Positive) is the number of non-interacting residues that are falsely predicted as interacting, TN(True Negative) denotes the number of non-interacting sites that are identified correctly, and FN(False Negative) denotes the number of interacting sites that are identified falsely. Because all those five metrics are threshold-dependent, we also utilize the area under the precision-recall curve (AUC PR) which gives a threshold-independent evaluation on the overall performance. Also, AUC PR is sensitive on imbalanced data, and PPIs prediction is normally an imbalanced learning problem(Zeng et al., 2020). Therefore, we take AUC PR as the most import metric for model evaluation(Staeheli and Mitchell, 2006).

2.5 Implementation

We implement our model using PyTorch v1.4(Steiner et al., 2019). The training configurations are: loss function: cross-entropy loss, optimization: Adaptive Momentum (Adam); learning rate: 0.1, 0.01, 0.001; batch size: 32, 64, 128; dropout: 0.2, 0.5, 0.7; sliding window length: 3, 5, 7, 9, 11. Convolution layers with the kernels (3, 5 and 7) are used, and the convolutional results are concatenated and fed to a two-layer fully connected networks. The first fully connected layer has 512 nodes and the second fully connected layer has 256 nodes. An independent validation set (10% of our training set) is used to tune parameters. Training time of each epoch varies roughly from 5 to10 minutes depending on the sliding window length, using a single NVIDIA RTX2080 GPU.

3 Results

3.1 Comparison with competing methods

To evaluate the performance of our model for PPIs prediction, we compared it with six competing sequenced-based computational methods including PSIVER(Murakami and Mizuguchi, 2010), SPPIDER(Sanchez, 2006), SPRINGS(Singh G, Dhole K, Pai PP, 2014), ISIS(Ofran and Rost, 2007), RF_PPI(Hou et al., 2017) and DeepPPISP(Zeng et al., 2020). It must be noted that DeepPPISP(Zeng et al., 2020) uses not only protein sequence but also structure information when it computes protein secondary structure using DSSP(Kabsch and Sander, 1983) which takes protein structure as input.

The six competing methods all employ a sliding window approach. Except for DeepPPISP utilizing a deep learning framework, all others use shallow machine learning methods. The sequenced-based features in those models include PSSM and predicted properties such as solvent accessibility, fingerprints, evolutionary information, and structural information. Our model uses one-hot encoding of protein sequence, PSSM, and predicted accessibility, secondary structure information and backbone dihedral angles. Specially, DeepPPISP combines local features with global feature from the whole protein sequence. Our model only utilizes local features from residues within a sliding window length.

Table 2 shows the results of our model and six competing methods on the testing set. Results of PSIVER, SPPIDER, SPRINGS, ISIS, RF_PPI and DeepPPISP are excerpted from DeepPPISP (Zeng et al., 2020). Our model outperforms other competing methods on most assessment metrics. Although our model's accuracy is slightly lower than ISIS, our method achieves the highest scores on all other metrics.

Table 2. Performance of our model and other competing methods

Method	AUC PR	ACC	Precision	Recall	F-measure	MCC
PSIVER	0.250	0.653	0.253	0.468	0.328	0.138
SPPIDER	0.230	0.622	0.209	0.459	0.287	0.089
SPRINGS	0.280	0.631	0.248	0.598	0.350	0.181
ISIS	0.240	0.694	0.211	0.362	0.267	0.097
RF_PPI	0.210	0.598	0.173	0.512	0.258	0.118
DeepPPISP	0.320	0.655	0.303	0.577	0.397	0.206
Our Method	0.359	0.657	0.313	0.611	0.414	0.229

3.2 The effects of different sliding window lengths

We employ the sliding window approach with different lengths (i.e. 3, 5, 7, 9, 11). As shown in Table 3, AUC PR, Precision, Recall, F-measure and MCC obtained by our model of a sliding window length 5 are 0.359, 0.657, 0.313, 0.414 and 0.229, respectively, which are better than all other window lengths. Recall of window length of 5 is lower than window lengths of 3, 7 and 11. However, Precision and Recall are usually contradictory performance metrics. The higher Precision is always accompanied by the lower Recall. Therefore, the results show that our method performs best when the sliding window length is 5.

Table 3. Performance of our model with different sliding window length

Window Length	AUC PR	ACC	Precision	Recall	F-measure	MCC
3	0.346	0.611	0.288	0.654	0.399	0.204
5	0.359	0.657	0.313	0.611	0.414	0.229
7	0.339	0.630	0.300	0.652	0.411	0.222
9	0.321	0.623	0.287	0.610	0.390	0.190
11	0.308	0.548	0.264	0.717	0.386	0.179

Compared with DeepPPISP(Zeng et al., 2020), our model doesn't use global features, and still achieves better performance with a small sliding window size. It also shows that too long or too short sliding window length isn't helpful for PPIs prediction, and an appropriate number of neighboring residues is effective for better representation of target residue's local environment. When the sliding window length is 3, the performance of our model is poor as there are too few neighboring residues to get effective information of target residue for prediction. When the sliding window length is too long, such as 11, too many neighboring residues cause distraction of "attention". Therefore, our model performs badly as well.

picture(0,0)(-35,0)(1,0)30 (0,-35)(0,1)30 picture picture(0,0)(35,0)(-1,0)30 (0,-35)(0,1)30 picture

bioRxiv preprint doi: https://doi.org/10.1101/2021.07.10.451856; this version posted July 12, 2021. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under a CC-BY-NC-ND 4.0 International license.

picture(0,0)(-35,0)(1,0)30 (0,35)(0,-1)30 picture picture(0,0)(35,0)(-1,0)30 (0,35)(0,-1)30 picture

3.3 The effects of different feature types

To find out what role each feature type plays in our model, we compare the performances of our model with a sliding window length of 5 using different feature combinations. Table 4 shows the effects of different input feature types in our model. AUC PR drops to 0.347, 0.303, 0.353, 0.279, and 0.339 from 0.359, when the input features of our model miss One-hot, PSSM, SS8, ASA and RSA, and backbone dihedral angles(ϕ and ψ), respectively. Among them, input features missing ASA and RSA have the largest drop. When the missing feature is ASA and RSA, ACC, Precision, Recall, F-measure, MCC drop to 0.544, 0.248, 0.641, 0.358 and 0.129, respectively. Except for ACC, the missing of ASA and RSA cause largest drop on other metrics. It indicates that ASA and RSA are the most import features in our model.

Table 4. Performance of our model with different features combination

Missing Features	AUC PR	ACC	Precision	Recall	F-measure	MCC
One_hot	0.347	0.607	0.292	0.693	0.411	0.223
PSSM	0.303	0.521	0.260	0.769	0.388	0.185
SS8	0.353	0.567	0.280	0.757	0.409	0.221
ASA,RSA	0.279	0.544	0.248	0.641	0.358	0.129
ϕ, ψ	0.339	0.620	0.293	0.651	0.404	0.212
None	0.359	0.657	0.313	0.611	0.414	0.229

In the state-of-the-art method DeepPPISP(Zeng et al., 2020), the raw protein sequence, PSSM, secondary structure are used. And, inspired by word embedding techniques in natural language processing, an embedding layer is applied to transform the one-hot encoding of a raw protein sequence to a denser vector. Therefore, raw protein sequence contributes most in DeepPPISP(Zeng et al., 2020). In our study, the usage of new predicted features provides more useful information. And the same conclusion is that all features used can get best results.

3.4 The effect and visualization of attention mechanism

Table 5 shows our attention-based convolutional neural network model significantly outperforms the CNNs model on most evaluation metrics due to the attention mechanism. As expected, the attention mechanism distinguishes different effect of each neighboring residue and provides more accurate information about target residue's local environment.

Table 5. Effect of attention mechanism

Method	AUC PR	ACC	Precision	Recall	F-measure	MCC
CNNs	0.350	0.641	0.304	0.631	0.410	0.222
Our Method	0.359	0.657	0.313	0.611	0.414	0.229

Fig. 2 shows the averaged attention score visualization of each residue type pair with various sliding window sizes. The vertical axis represents the types of target amino acid residues and the horizontal axis represents the neighboring residues. All residues types are in alphabetical order. The red color represents that more attention is put on the corresponding neighboring residue. On the contrary, the green color represents that less attention is put. For example, when the sliding window length is 11, and the target residue is Cysteine(C), the residue Tryptophan(W) draws the most attention among all neighboring residue types.

The results show that the variations in attention scores don't correlate with the properties of amino acids such as polarity, charge and alkalinity or acidity. In PPIs prediction, the attention mechanism captures context information of protein sequence other than properties of individual amino acid.

Fig. 2. Averaged attention scores visualization of each residue type pair with various sliding window lengths.

picture(0,0)(-35,0)(1,0)30 (0,-35)(0,1)30 picture picture(0,0)(35,0)(-1,0)30 (0,-35)(0,1)30 picture

bioRxiv preprint doi: https://doi.org/10.1101/2021.07.10.451856; this version posted July 12, 2021. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under a CC-BY-NC-ND 4.0 International license.

picture(0,0)(-35,0)(1,0)30 (0,35)(0,-1)30 picture picture(0,0)(35,0)(-1,0)30 (0,35)(0,-1)30 picture

However, it is obvious that when the sliding window length is 3, there are very few dark red and dark green cells. In other words, neighboring residues don't get specific and effective attention as the window length is too short. From both Table 3 and Fig. 2, we can see that when the sliding window length is 5, our model performs best and neighboring residues get most specific and effective attention. And when the length of the sliding window grows longer than 5, the averaged attention scores become even. Especial when the sliding window length is 11, almost every neighboring residue doesn't get enough "attention". When the sliding window length is 9, "attention" focuses on a few types of neighboring residues such as Proline(P) which cause poor results. We count the number of occurrences of each residue type which is not relevant to this concentration of "attention". This situation needs further study.

4 Conclusion

In this study, we combine attention mechanism with convolutional neural networks to capture the residue correlations within a sliding window for PPIs prediction using protein sequence information. Our model can distinguish the effects of different neighboring residues on a target residue. It captures more accurate information of the target residue's local environment. The experiments show that our model significantly outperforms other state-of-the-art methods.

Though our method has superior performance over other competing methods, it also has some potential limitations. First, similar like other sequence-base methods, our program takes a lot of time to generate sequence profiles by running PSI-BLSAT(Altschul et al., 1997) and NetsurfP-2.0(Klausen et al., 2019). Second, although the attention mechanism improves the overall performance, it takes additional computation consumption to calculate attention scores, especially when the sliding window length is long. Third, with the sliding window length growing much longer, teh attention mechanism doesn't lead to better results.

In this study, we show that paying different attention on different neighboring residue can be helpful for sequence-based PPIs prediction. We believe that the attention mechanism has a great potential in other biological sequence analysis and prediction problems. Currently we only employ attention mechanism by a one-layer fully connected network. More complicated neural networks architecture should be used for seeking better results.

However, amino acid residues that are far apart in the sequence may be adjacent in space and longer sliding window length needs to be considered. The results of our model show than too long sliding window length cause the concentration and distraction of "attention". This problem needs further study and discussion.

Acknowledgements

We thank Dr. Zhang and Dr. Nan (Zhengzhou University) for constructive discussions. The authors thank anonymous reviewers for valuable suggestions and comments, which greatly improve this article.

Funding

This work was supported by Bingtuan Science and Technology Project (2019AB034), 'Created Major New Drugs' of Major National Science and Technology (No. 2019ZX09301159), and Leading Talents Fund in Science and Technology Innovation in Henan Province(194200510002).

Conflict of Interest: none declared.

References

Altschul,S.F. *et al*. (1997) Gapped BLAST and PSI-BLAST: a new generation of protein database search programs. *Nucleic Acids Res.*, **25**, 33893402.

Bahdanau,D. *et al*. (2015) Neural machine translation by jointly learning to align and translate. In, *International Conference on Learning Representations*

Berkeley,U.C.*et al*. (2020) Transformer Protein Language Models Are Unsupervised Structure Learners. *bioRxiv*, 1–24.

Brettner,L.M. and Masel,J. (2012) Protein stickiness, rather than number of functional protein-protein interactions, predicts expression noise and plasticity in yeast. *BMC Syst. Biol.*, **6**, 1–10.

Chen,K. and Kurgan,L. (2009) Investigation of atomic level patterns in protein-small ligand interactions. *PLoS One*, **4**, 1-14.

Chen,L. *et al*. (2020) TransformerCPI: Improving compound–protein interaction prediction by sequence-based deep learning with self-attention mechanism and label reversal experiments. *Bioinformatics*, **36**, 4406–4414.

Dhole,K. *et al*. (2014) Sequence-based prediction of protein-protein interaction sites with L1-logreg classifier. *J. Theor. Biol.*, **348**, 47–54.

Ding,X. *et al*. (2010) Computational Prediction of DNA-Protein Interactions: A Review. *Curr. Comput. Aided-Drug Des.*, **6**, 197–206.

Fornes,O. *et al*. (2014) On the use of knowledge-based potentials for the evaluation of models of protein-protein, protein-DNA, and protein-RNA interactions. In, *Advances in Protein Chemistry and Structural Biology.*, Academic Press Inc., Vol. 94, pp. 77–120.

Hou,Q. *et al*. (2017) Seeing the trees through the forest: Sequencebased homo- and heteromeric protein-protein interaction sites prediction using random forest. *Bioinformatics*, **33**, 1479–1487.

Kabsch,W. and Sander,C. (1983) Dictionary of protein secondary structure: Pattern recognition of hydrogen-bonded and geometrical features. *Biopolymers*, **22**, 2577–2637.

Karimi,M. *et al*. (2019) DeepAffinity: Interpretable deep learning of compound-protein affinity through unified recurrent and convolutional neural networks. *Bioinformatics*, **35**, 3329–3338.

Klausen,M.S. *et al*. (2019) NetSurfP-2.0: Improved prediction of protein structural features by integrated deep learning. *Proteins Struct. Funct. Bioinforma.*, **87**, 520–527.

Kozma,D. *et al*. (2013) PDBTM: Protein data bank of transmembrane proteins after 8 years. *Nucleic Acids Res.*, **41**, 524–529.

Kuzmanov,U. and Emili,A. (2013) Protein-protein interaction networks: Probing disease mechanisms using model systems. *Genome Med.*, **5**, 1–12.

Kyriakis,J.M. and Avruch,J. (2001) Mammalian mitogen-activated protein kinase signal transduction pathways activated by stress and inflammation. *Physiol. Rev.*, **81**, 807–869.

Laskowski,R.A. (2009) PDBsum new things. *Nucleic Acids Res.*, **37**, 355–359.

Li,Y. and Ilie,L. (2020) DELPHI: Accurate deep ensemble model for protein interaction sites prediction. *Bioinformatics*, **37**, 896–904.

McGinnis,S. and Madden,T.L. (2004) BLAST: At the core of a powerful and diverse set of sequence analysis tools. *Nucleic Acids Res.*, **32**, 20–25.

Mian,I.S. *et al*. (1991) Structure, function and properties of antibody binding sites. *J. Mol. Biol.*, **217**, 133–151.

Murakami,Y. and Mizuguchi,K. (2010) Applying the Naïve Bayes classifier with kernel density estimation to the prediction of protein-protein interaction sites. *Bioinformatics*, **26**, 1841–1848.

Nibbe,R.K. *et al*. (2011) Protein-protein interaction networks and subnetworks in the biology of disease. Wiley Interdiscip. *Rev. Syst. Biol. Med.*, **3**, 357–367.

Ofran,Y. and Rost,B. (2007) ISIS: Interaction sites identified from sequence. *Bioinformatics*, **23**, 13–16.

Ofran,Y. and Rost,B. (2003) Predicted protein-protein interaction sites from local sequence information. *FEBS Lett.*, **544**, 236–239.

Orii,N. and Ganapathiraju,M.K. (2012) Wiki-Pi: A Web-Server of Annotated Human Protein-Protein Interactions to Aid in Discovery of Protein Function. *PLoS One*, **7**, e49029.

Petta,I. et al. (2016) Modulation of protein-protein interactions for the development of novel therapeutics. *Mol. Ther.*, **24**, 707–718.

Pittala,S. and Bailey-Kellogg,C. (2020) Learning context-aware structural representations to predict antigen and antibody binding interfaces. *Bioinformatics*, **36**, 3996–4003.

Ragoza,M. *et al*. (2017) Protein-Ligand Scoring with Convolutional Neural Networks. *J. Chem. Inf. Model.*, **57**, 942–957.

Rohani,M.M. and Zydney,A.L. (2009) Effect of surface charge distribution on protein transport through semipermeable ultrafiltration membranes. *J. Memb. Sci.*, **337**, 324–331.

Sanchez,G. (2006) Prediction-Based Fingerprints of Protein–Protein Interactions. *PROTEINS Struct. Funct. Bioinforma.*, **645**, 630–645.

Singh G, Dhole K, Pai PP,M.S. (2014) SPRINGS: Prediction of Protein-Protein Interaction Sites Using Artificial Neural Networks. *J. Proteomics Comput. Biol.*,

picture(0,0)(-35,0)(1,0)30 (0,-35)(0,1)30 picture picture(0,0)(35,0)(-1,0)30 (0,-35)(0,1)30 picture

bioRxiv preprint doi: https://doi.org/10.1101/2021.07.10.451856; this version posted July 12, 2021. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under a CC-BY-NC-ND 4.0 International license.

picture(0,0)(-35,0)(1,0)30 (0,35)(0,-1)30 picture picture(0,0)(35,0)(-1,0)30 (0,35)(0,-1)30 picture

1, 01–07.

Staeheli,L.A. and Mitchell,D. (2006) The Relationship Between Precision-Recall and ROC Curves Jesse. In, *International Conference on Machine Learning.*, pp. 233–240.

Steiner,B. *et al*. (2019) PyTorch: An Imperative Style, High-Performance Deep Learning Library. In, *Conference on Neural Information Processing Systems.*

Sudha,G. *et al*. (2014) An overview of recent advances in structural bioinformatics of protein-protein interactions and a guide to their principles. *Prog. Biophys. Mol. Biol.*, **116**, 141–150.

Sussman,J.L. *et al*. (1999) The protein data bank: Bridging the gap between the sequence and 3D structure world. *Genetica*, **106**, 149–158.

Terentiev,A.A. *et al*. (2009) Dynamic proteomics in modeling of the living cell. Protein-protein interactions. *Biochem.*, **74**, 1586–1607.

Wardah,W. *et al*. (2020) Predicting protein-peptide binding sites with a deep convolutional neural network. *J. Theor. Biol.*, **496**, 110278.

Wei,Z. Sen *et al*. (2015) A Cascade Random Forests Algorithm for Predicting Protein-Protein Interaction Sites. *IEEE Trans. Nanobioscience*, **14**, 746–760.

Wei,Z. Sen *et al*. (2016) Protein-protein interaction sites prediction by ensembling SVM and sample-weighted random forests. *Neurocomputing*, **193**, 201–212.

Wodak,S.J. *et al*. (2013) Protein-protein interaction networks: The puzzling riches. *Curr. Opin. Struct. Biol.*, **23**, 941–953.

Zeng,M. *et al*. (2020) Protein-protein interaction site prediction through combining local and global features with deep neural networks. *Bioinformatics*, **36**, 1114–1120.

Zhang,F. *et al*. (2020) PROBselect: Accurate prediction of protein-binding residues from proteins sequences via dynamic predictor selection. *Bioinformatics*, **36**, I735–I744.

Zhang,J. and Kurgan,L. (2019) SCRIBER: Accurate and partner type-specific prediction of protein-binding residues from proteins sequences. *Bioinformatics*, **35**, i343–i353.

Zhao,Z. and Wu,Y. (2016) Attention-based convolutional neural networks for sentence classification. In, *Proceedings of the Annual Conference of the International Speech Communication Association*, INTERSPEECH., pp. 705–709.

picture(0,0)(-35,0)(1,0)30 (0,-35)(0,1)30 picture picture(0,0)(35,0)(-1,0)30 (0,-35)(0,1)30 picture