Bioinformatics, YYYY, 0–0
doi: 10.1093/bioinformatics/xxxxx
Advance Access Publication Date: DD Month YYYY
Manuscript Category

Subject Section

Protein-protein interaction site prediction through combining local and global features with deep neural networks

Min Zeng[1], Fuhao Zhang[1], Fang-Xiang Wu[2], Yaohang Li[3], Jianxin Wang[1] and Min Li[1,*]

[1]School of Computer Science and Engineering, Central South University, Changsha, 410083, P.R. China, [2]Division of Biomedical Engineering and Department of Mechanical Engineering, University of Saskatchewan, Saskatoon, SKS7N5A9, Canada, [3]Department of Computer Science, Old Dominion University, Norfolk, VA 23529, USA.

*To whom correspondence should be addressed.

Associate Editor: XXXXXXX

Received on XXXXX; revised on XXXXX; accepted on XXXXX

Abstract

Motivation: Protein-protein interactions (PPIs) play important roles in many biological processes. Conventional biological experiments for identifying PPI sites are costly and time-consuming. Thus, many computational approaches have been proposed to predict PPI sites. Existing computational methods usually use local contextual features to predict PPI sites. Actually, global features of protein sequences are critical for PPI site prediction.

Results: A new end-to-end deep learning framework, named DeepPPISP, through combining local contextual and global sequence features, is proposed for PPI site prediction. For local contextual features, we use a sliding window to capture features of neighbors of a target amino acid as in previous studies. For global sequence features, a text convolutional neural network is applied to extract features from the whole protein sequence. Then the local contextual and global sequence features are combined to predict PPI sites. By integrating local contextual and global sequence features, DeepPPISP achieves the state-of-the-art performance, which is better than the other competing methods. In order to investigate if global sequence features are helpful in our deep learning model, we remove or change some components in DeepPPISP. Detailed analyses show that global sequence features play important roles in DeepPPISP.

Availability: The DeepPPISP web server is available at http://bioinformatics.csu.edu.cn/PPISP/. The source code can be obtained from https://github.com/CSUBioGroup/DeepPPISP.

Contact: limin@mail.csu.edu.cn

Supplementary information: Supplementary data are available at *Bioinformatics* online.

1 Introduction

Proteins perform functions in various biological processes, and they rarely act alone as their functions tend to be regulated (Han, et al., 2004). Protein-protein interactions (PPIs) are the physical contacts between two or more proteins and are crucial for the function of proteins (De Las Rivas and Fontanillo, 2010; Li, et al., 2019). Identification of PPI sites can help understand how a protein performs its biological functions (Li, et al., 2018). In addition, it can help design new antibacterial drugs (Russell and Aloy, 2008). Conventional biological experimental methods, such as two-hybrid screening and affinity purification coupled to mass spectrometry, are used to identify PPIs (Brettner and Masel, 2012; Terentiev, et al., 2009; Wodak, et al., 2013). However, these biological experimental methods are costly and time-consuming. Thus, developing

an accurate computational approach to predict PPI sites would be of great value to biologists.

In the past two decades, a lot of computational approaches have been established to predict PPI sites. These methods can be roughly divided into three categories: protein–protein docking and modeling, structure-based and sequence-based methods (Hou, et al., 2017). Protein–protein docking and structure-based methods usually need structural details (Hou, et al., 2016), while many proteins have no structural information except for their protein sequences. In addition, with the rapid development of high-throughput sequencing techniques, a growing number of protein sequences can be obtained, which makes sequence-based methods get more attention. A majority of computational methods employ machine learning algorithms, including shallow neural networks (Chen and Zhou, 2005; Fariselli, et al., 2002; Ofran and Rost, 2003; Porollo and Meller, 2007), support vector machine (Li, et al., 2008; Sriwastava, et al., 2015; Yan, et al., 2004), random forest (Hou, et al., 2017; Northey, et al., 2017; Wang, et al., 2018), Naïve Bayes (Lin and Chen, 2013), ensemble learning (Deng, et al., 2009), and conditional random field (Li, et al., 2007). In these studies, a large number of features extracted from protein sequences are used. The commonly used features are evolutionary information (Caffrey, et al., 2004; Carl, et al., 2008; Choi, et al., 2009), secondary structure (Guharoy and Chakrabarti, 2007; Li, et al., 2012; Ofran and Rost, 2007). In addition to these commonly used features, some other physiochemical, biophysical and statistical features, e.g. accessible surface area (de Vries and Bonvin, 2008; Hou, et al., 2017), protein size (Martin, 2014), backbone flexibility (Bendell, et al., 2014), and sequence specificity (Hou, et al., 2015), are used for PPI site prediction.

It is well known that local contextual features are crucial for PPI site prediction. Thus, many computational methods used a sliding window-based method to extract features of neighbors of an amino acid. The sliding window-based method is not only used to extract local features of the target amino acid in PPI site prediction (Hou, et al., 2017; Mihel, et al., 2008; Wang, et al., 2018), but also used in various protein-related problems including protein structure prediction and protein disorder prediction (Yaseen and Li, 2013). In addition, global features of protein sequences also hold vital evidence for the prediction of PPI sites. The previous studies have reported that global sequence features are helpful to predict interface amino acids (Yan, et al., 2004). The existing computational methods have achieved good performance, but they do not take global sequence features into account in their model. Actually, the lack of global sequence features can decrease the performance of machine learning algorithms.

To extract and integrate global sequence features, we use deep learning techniques. In recent years, deep learning techniques have been successfully applied in bioinformatics (Li, et al., 2018; Li and Yu, 2016; Pan and Shen, 2018; Zeng, et al., 2019; Zhang, et al., 2019). Convolutional neural networks (CNNs) and recurrent neural networks (RNNs) are applied to extract sequence features and have proven to be effective in many biological tasks (Zeng, et al., 2018). Inspired by their success, we propose a deep learning model called DeepPPISP to predict PPI sites. The key idea of DeepPPISP is to extract not only local contextual features but also global features from protein sequences and integrate them into a deep learning framework. For local contextual features of a target amino acid, we use a sliding window to obtain the features of neighboring amino acids. For global features of entire protein sequences, we combine different deep learning structures, e.g. fully connected layers, CNNs, to extract sequence features. After the part of local contextual and global sequence feature extraction, two feature vectors are concatenated together to carry out the classification task. To our knowledge, it is the first time to apply deep learning techniques to combine local and global features from protein sequences to PPI site prediction.

To evaluate the performance of DeepPPISP, we compare DeepPPISP with 5 competing computational methods (PSIVER, SPPIDER, SPRINGS, ISIS and RF_PPI). The results show that DeepPPISP achieves the state-of-the-art performance for predicting PPI sites. In order to investigate whether the global sequence features are helpful to predict PPI sites, we remove or change some components in our model. The detailed analyses show that the global sequence features are very important in DeepPPISP.

2 Methods

2.1 Datasets

Similar to previous studies, we used the three benchmark datasets, i.e., Dset_186, Dset_72 (Murakami and Mizuguchi, 2010) and PDBset_164 (Singh, et al., 2014). Dset_186 has been built from the PDB database and consists of 186 protein sequences with the resolution less than 3.0 Å with sequence homology less than 25%. Dset_72 and PDBset_164 are constructed as the same as Dset_186. Dset_72 has 72 protein sequences and PDBset_164 consists of 164 protein sequences. These protein sequences in the three benchmark datasets have been annotated. Thus, we have 422 different annotated protein sequences. We remove two protein sequences as they do not have the definition of secondary structure of proteins (DSSP) file. In this study, an amino acid is defined as an interaction site if its absolute solvent accessibility is less than 1 $Å^2$, before and after the binding of a protein in the binding form; otherwise, it is defined as a non-interaction site. We count the number of interaction sites and non-interaction sites. 1923, 5517 and 6096 amino acids are interaction sites in Dset_186, Dset_72, and PDBset_164, respectively; 16217, 30702 and 27585 amino acids are non-interaction sites in Dset_186, Dset_72, and PDBset_164, respectively. Although these protein sequences in three datasets are not repeated, three datasets come from different research groups. To ensure that the training set and test set are from an identical distribution, we integrate three datasets to a fused dataset. We count the lengths of all sequences in the fused dataset. Table 1 shows the distribution of the lengths of all sequences. Then we divide the fused dataset into a training set (about 83.3% of randomly selected protein sequences) and a test set (the remaining protein sequences). Another advantage of doing this is that we can make full use of these protein sequences to train our deep learning model. As we know, training a deep learning model requires a large number of samples. Last, there are 350 protein sequences in the training set (50 proteins for independent validation set) while there are 70 protein sequences in the test set.

Table 1. Statistics of lengths of all sequences in the study.

Length range	1-100	100-200	200-300	300-400	400-500	500-600	600-700	700+
Number	85	176	68	56	23	7	4	3

2.2 Input features

The feature selection is a crucial step in a deep learning framework. As mentioned above, evolutionary information and secondary structure properties are used to encode features of each amino acid in protein sequences. In addition, the raw protein sequences are used in this study. These features are described in detail as follows.

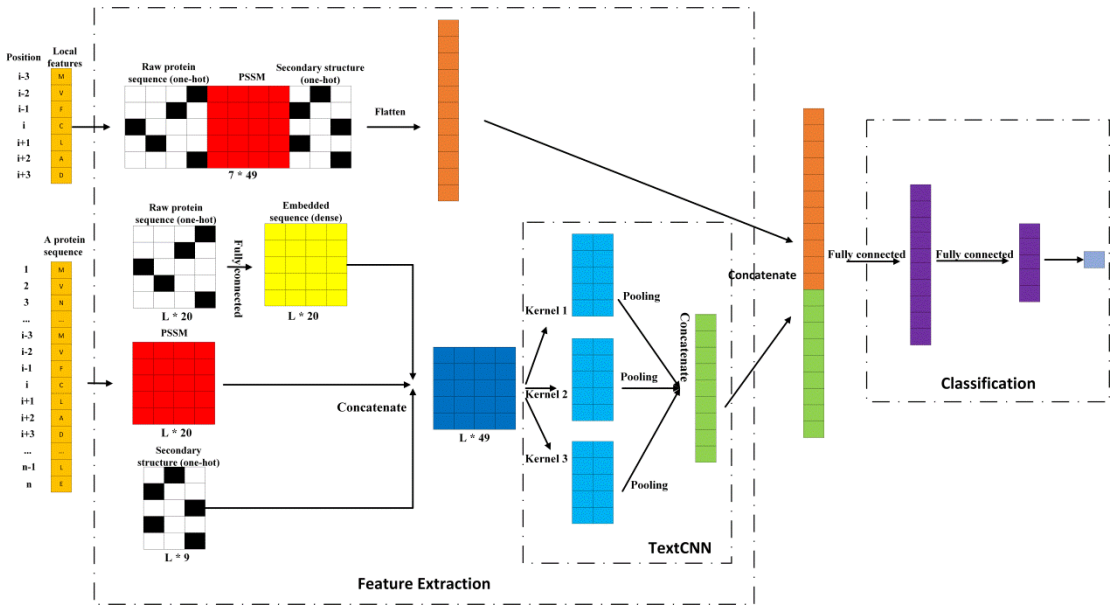

Fig.1. The deep neural network structure of DeepPPISP. The input consists of two types of data: subsequence and whole protein sequence. For subsequence, a sliding window size of 7 is applied to extract features of neighbors of a target amino acid. Then the combination of three types of features is flattened a local feature vector. For the whole protein sequence, the length (L) is set to 500. First, through raw sequence feature embedding, three types of features are concatenated to a preprocessed vector, which is fed into a TextCNN with different kernels (13, 15, and 17) to obtain a global sequence feature vector. The local and global feature vectors are concatenated, and then the concatenated vector is fed into two fully connected layers for prediction.

2.2.1 Position specific scoring matrix (PSSM)

The evolutionary information in PSSM has been proven to be effective in PPI site prediction. PSSM is generated by running the PSI-BLAST algorithm to search against the NCBI's Non-Redundant (NR) sequence database with three iterations and an E-value threshold of 0.001. Each amino acid is encoded as a vector with 20 elements that represent the probabilities of 20 amino acids occurring at this position.

2.2.2 Secondary structure

The secondary structure is a very popular feature used to encode structure information of amino acids in PPI site prediction. Secondary structure information is generated by running the DSSP program. We use eight-category secondary structure states (3_{10}−helix (G), α−helix (H), π−helix (I), β−strand (E), β−bridge (B), β−turn (T), bend (S) and loop or irregular (L)). Considering that some amino acids do not have their secondary structure states in the DSSP file. We use a 9-dimensional one-hot vector to encode them, i.e., only one element is one and the others are zero. The first 8 dimensions represent the state of each amino acid, and the last dimension represents no information about secondary structure states.

2.2.3 Raw protein sequences

Raw protein sequences can accurately represent each amino acid with its position. Most proteins consist of 20 types of different amino acids. Thus, we use a 20-dimensional one-hot vector to encode the types of amino acids in the protein.

Thus, using the three types of features, we obtain a 49-dimensional feature vector for each amino acid in protein sequences.

2.3 Network architecture and feature embedding

As illustrated in Fig.1, DeepPPISP consists of two parts, i.e., feature extraction and classification parts. DeepPPISP is an end-to-end model. The inputs to DeepPPISP are two types of features, local contextual and global sequence features. The feature extraction part is responsible for preprocessing and extracting useful local and global features and patterns to predict PPI sites.

For local contextual features, similar to previous studies, a sliding window-based method is applied to extract features of neighbors of an amino acid. Specifically, a sliding window size of (2n+1) means we consider the target amino acid at the center and 2n neighbor amino acids as input features of the target amino acid. For example, if sliding window size is 7, for each amino acid at position i, the features of amino acids at position i-3, i-2, i-1, i, i+1, i+2, i+3 are considered as its local contextual features. For those amino acids which do not have neighbors of amino acids in the left or right window, we use the all-zero vector of the same length as the feature vector as its missing features.

Global features of protein sequences are the focus of our study. For global sequence features, we applied deep learning techniques to learn them from protein sequences. Given a protein sequence A = a_1, a_2, … a_n, each amino acid is represented by a 49-dimensional feature vector (20-dimension for PSSM, 9-dimension for secondary structure, 20-dimension for raw protein sequences). From Table 1, we know that only 14 protein sequences have more than 500 amino acids. Thus the length of all protein sequences is normalized to 500. If a protein sequence longer than 500, then we truncate it; if shorter than 500, we pad it with zeros. In addition, we note that both the secondary structure vector and the raw

protein sequence vector are sparse one-hot vectors, while the PSSM vector is a dense vector. To avoid the inconsistency of different types of input features, inspired by word embedding techniques in natural language processing, an embedding layer is applied to transform sparse a raw protein sequence vector to a denser vector. The embedding layer is implemented as a fully connected layer. After the embedding layer, an embedded raw protein sequence vector is concatenated with the PSSM vector and the secondary structure vector as a preprocessed vector. After that, a text convolutional neural network with a max pooling layer is applied to extract the global features of the preprocessed vector. The output vectors of this layer are concatenated together as the global features of the input protein sequence.

The classification part consists of two fully connected layers and an output layer. In the classification part, there are two fully connected layers taking the concatenated vector as input. The output from the second fully connected layer is fed into the output layer with a sigmoid activation function, which performs binary classification to determine if the input amino acid is an interaction site.

2.4 Text convolutional neural networks

In addition to local contextual features, global sequence features are crucial in PPI site prediction. Text convolutional neural networks (TextCNN) can capture global features of protein sequences. Traditional CNNs are usually used to extract features of two-dimensional image data. In recent years, some searchers started to use CNNs to address texts. The central idea is that a text can be treated as a one-dimensional image. Thus, one-dimensional CNNs can be used to capture the relationship between adjacent words. Inspired by this, we treat the whole protein sequence as a text in order to better extract features of the whole protein sequence by using TextCNN. Specifically, assume that a protein sequence consists of n amino acids, and each amino acid is represented by an m-dimensional vector. Then the protein sequence can be treated as an image, i.e., the width is n, the height is 1, and the channel is m. To capture features of different lengths of subsequence, multiple different scale convolutional kernels are used. We can use different scale convolutional kernels to obtain the relationship between different numbers of adjacent amino acids (Zeng, et al., 2019). A max pooling layer is applied to capture the most important features of each channel and reduce the dimension of the output vector. Then the output vectors of a max pooling layer are concatenated together as a concatenated vector which contains global features of the whole protein sequence. Supplementary Fig.S1 gives the illustration of TextCNN.

2.5 Applicability domain

In DeepPPISP, three physicochemical or topological properties are used to define the applicability domain, which are putative relative solvent accessibility (RSA) score, polarity and protein sequence length. RSA score is a key property relevant to the characterization of PPI binding sites and is predicted by ASAquick (Zhang and Kurgan, 2017). Each residue in different position has its own unique score. We calculate the average RSA score for each protein in the training set. Similarly, the polarity is relevant to PPI and is quantified using the AAindex resource (Zhang and Kurgan, 2019). Each type of residues has its own polarity. We calculate the average polarity for each protein in the training set. Protein sequence length is the number of residues in the protein. We analyze the distribution of these properties in our training set. The definition is as follows: if a specific physicochemical or topological property in the range of 5% to 95% of the training set, it regarded as in domain; if it is in the range of 0% to 5% or 95% to 100%, it is regarded as warning domain; if it is higher than the maximum value or lower than the minimum value, it is regarded as out domain. Supplementary Fig. S2 gives the definition of the applicability domain of three physicochemical or topological properties.

2.6 Evaluation metrics

For PPI site prediction, we assume the interaction sites to represent the positive samples and non-interaction sites to represent the negative samples. To evaluate the performance of our model and other methods in PPI sites, six evaluation metrics are used in this study: accuracy (ACC), precision, recall, F-measure, area under the receiver operator characteristic curve (AUC), area under the precision-recall curve (auPR) and the Matthews correlation coefficient (MCC).

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (1)$$

$$precision = \frac{TP}{TP+FP} \quad (2)$$

$$recall = \frac{TP}{TP+FN} \quad (3)$$

$$F-measure = \frac{2*precision*recall}{precision+recall} \quad (4)$$

$$MCC = \frac{TP*TN-FP*FN}{\sqrt{(TP+FP)(TP+FN)(TN+FP)(TN+FN)}} \quad (5)$$

where TP represents the number of interaction sites identified correctly, FN denotes the number of interaction sites identified incorrectly, FP represents the number of non-interaction sites identified incorrectly, TN denotes the number of non-interaction sites identified correctly. It should note that in an imbalanced learning problem, F-measure, MCC and auPR are three most important evaluation metrics as they can provide comprehensive measure than other evaluation metrics (Zeng, et al., 2016).

2.7 Implementation details

Our deep learning framework is implemented with PyTorch (http://pytorch.org/) which is a popular deep learning package. The loss function we used is the cross-entropy loss, defined as follows:

$$Loss = -\frac{1}{n}\sum [y\, log(y_{pred}) + (1-y)\, log(1-y_{pred})] \quad (6)$$

where n is the number of all training data, y is the true label, and y_{pred} is the predicted label. There are two properties to use the cross-entropy function as a loss function. First, it is non-negative. Second, if the predicted label is close to the true label for all training data, the function is close to zero.

The optimizer we used is Adaptive Momentum (Adam). DeepPPISP uses the following formula to update the weights:

$$\theta_{t+1} = \theta_t - \frac{\alpha}{\sqrt{\hat{v}_t}+\varepsilon}\,\hat{m}_t \quad (7)$$

where θ_{t+1} is the updated parameter, α is the learning rate, ε is a constant added to maintain numerical stability, $\hat{m}_t$ and $\hat{v}_t$ are bias-corrected first and second moment estimates, respectively.

To extract local contextual and global sequence features, the length of sliding window is set to 7; the length of protein sequence is set to 500. For the deep learning structure, the batch size is set to 64; the learning rate is set to 0.001; multi-scale CNN layers with the kernels (13, 15, and 17) are used to extract global sequence features in TextCNN. The first fully connected layer in the classification part has 1024 nodes and the

second fully connected layer has 256 nodes. To avoid overfitting, a dropout rate of 0.2 is applied in DeepPPISP. In this study, we use an independent validation set to tune parameters. The independent validation set is a hold out subset (50 proteins) of our training set. The workflow of training is that we train our model on the training set. Then we use independent validation set to tune the parameters of our model and see how well the model is. After having done this long enough, we can have a best model and evaluate it on the test set to get an unbiased estimate.

3 Results

3.1 Comparison with competing methods

To evaluate the performance of DeepPPISP in predicting PPI sites, we compared DeepPPISP with 5 competing methods (PSIVER, SPPIDER, SPRINGS, ISIS and RF_PPI). The 5 competing methods all used shallow machine learning methods as their predictors. PSIVER (Murakami and Mizuguchi, 2010) used sequence features (PSSM and predicted accessibility) to predict PPI sites by using a Naïve Bayes classifier. SPPIDER (Porollo and Meller, 2007) used alternative machine learning techniques which combine fingerprints with other sequence and structure information to predict PPI sites. SPRINGS (Singh, et al., 2014) utilized shallow neural network algorithm based on evolutionary information, averaged cumulative hydropathy and predicted relative solvent accessibility to predict PPI sites. ISIS (Ofran and Rost, 2007) used a shallow neural networks to combine predicted structural features with evolutionary information to predict PPI sites. RF_PPI (Hou, et al., 2017) was developed by Hou et al., which applied the random forest algorithm based on various features to predict PPI sites. These methods all used local contextual information but did not take into account global sequence features (see Supplementary Table S1).

Table 2. Predictive performance of DeepPPISP and other competing methods (PSIVER, SPPIDER, SPRINGS, ISIS and RF_PPI) on the test set.

Method	ACC	Precision	Recall	F-measure	MCC
PSIVER	0.653	0.253	0.468	0.328	0.138
SPPIDER	0.622	0.209	0.459	0.287	0.089
SPRINGS	0.631	0.248	0.598	0.350	0.181
ISIS	0.694	0.211	0.362	0.267	0.097
RF_PPI	0.598	0.173	0.512	0.258	0.118
DeepPPISP	0.655	0.303	0.577	0.397	0.206

Table 2 shows the results of DeepPPISP and 5 competing computational methods on the test set. From Table 2, we found that most of the assessment metrics obtained by DeepPPISP were higher than other competing methods. While accuracy and recall of DeepPPISP are lower than ISIS and SPRINGS, respectively, the other assessment metrics are higher than other competing methods. Precision, F-measure and MCC obtained by DeepPPISP are 0.303, 0.397 and 0.206, respectively, which are better than PSIVER (0.253, 0.328 and 0.138), SPPIDER (0.209, 0.287 and 0.089), SPRINGS (0.248, 0.350 and 0.181), ISIS (0.211, 0.267 and 0.097), and RF_PPI (0.173, 0.258 and 0.118). It is a remarkable fact that PPI site prediction is an imbalanced learning problem, thus we pay more attention to F-measure and MCC. The F-measure and MCC of DeepPPISP are the highest in the all existing methods, which shows that DeepPPISP outperforms the other computational methods. The PR curves of DeepPPISP and other competing methods are shown in Fig. 2. It demonstrates that auPR of DeepPPISP is higher than that of other competing methods. In addition, we performed training on the combination of Dset_186 and PDBset_164 datasets and testing on the Dset_72 dataset. Although the results of DeepPPISP on Dset_72 dataset are slightly lower than those of DeepPPISP on the test set, DeepPPISP performs better than other competing models. Detailed results can be found in Supplementary Tables S2 and Fig. S3. We also compared DeepPPISP with a structure-based method (IntPred) (see Supplementary Tables S3).

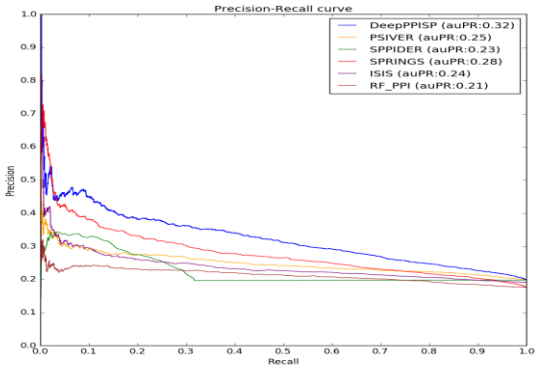

Fig. 2. PR curves of DeepPPISP and other competing methods on the test set.

3.2 The effects of different dimensions of the global sequence features

In this study, we focused on whether global sequence features are useful for predicting PPI sites. First, we tested the effects of global sequence features. We trained our model based on the raw training set but removing global sequence features extraction part. Table 3 shows the performance of our model by using local textual features to predict PPI sites. ACC, precision, recall, F-measure and MCC obtained without global sequence features are 0.520, 0.240, 0.663 0.353 and 0.118, respectively. Compared with the raw DeepPPISP model, all evaluation metrics are lower than DeepPPISP model except recall. Through combining local textual and global sequence features, we can get better performance than using the only local textual feature (about 4% higher in F-measure, 9% higher in MCC), which shows the advantages of global sequence features. Supplementary Fig. S4 shows the ROC curves of DeepPPISP and DeepPPISP without global sequence features. The AUC value of DeepPPISP is higher than DeepPPISP without global sequence features.

Table 3. Predictive performance of DeepPPISP and DeepPPISP without global sequence features.

Model	ACC	Precision	Recall	F-measure	MCC
Without global features	0.520	0.240	0.663	0.353	0.118
Global & local features	0.655	0.303	0.577	0.397	0.206

In addition to show the advantages of global sequence features, we changed different dimensions of global sequence features to observe the effects and variances of predictive performance. In DeepPPISP, the length of the local textual feature vector is 343 (7*49). We used different lengths of global sequence feature vector to train our model. Specifically, the different ratios of the length of global sequence feature vectors to the length of local textual feature vector (1:1, 1:2, 1:3, 1:4, 1:5, 2:1, 3:1, 4:1, 5:1) were applied to train DeepPPISP. The predictive performances are shown in Supplementary Table S4. From this table, we observed that the best performance was obtained with a ratio of 2:1. DeepPPISP obtains the accuracy of 65.5%, precision of 0.303, recall of 0.577, F-measure of 0.397, and MCC of 0.206, respectively, which are better than the other ratios. It can be also observed from Supplementary Table S4 that when we used the ratio of 1:5 and 5:1 for training, the results are very bad. In the ratio of 1:5, F-measure and MCC are 0.384 and 0.179, respectively; in the ratio of 5:1, F-measure and MCC are 0.386 and 0.182, respectively. The reason we believe is that the predictive performance decreases when using extremely unbalanced ratio. For instance, when using the ratios of 5:1, the global sequence feature vector dominated the whole feature vector. A protein sequence consists of hundreds of amino acids and these amino acids share a vector of global sequence features. Thus, the predictive results are mainly determined by the global sequence features and are biased to a certain result. With the ratio of 1:5, the local contextual feature vector dominated the whole feature vector and the global sequence feature vector is not so important. If two amino acids are far apart (more than 7 amino acids in length), they do not have the same amino acid in their neighbors and do not share local features. Thus, the predictive results are mainly determined by the local contextual features and vary a lot.

3.3 The effects of different types of input features

Besides the dimension of the global sequence features, the different types of input features (raw protein sequences, PSSM, secondary structure) play different roles in our model. In order to discover what role each type of feature plays in DeepPPISP, we conducted an ablation study by removing each feature in DeepPPISP. Specifically, we compared the performances of different models without raw protein sequences, PSSM, or secondary structure. From the results presented in Supplementary Table S5, it is clear that the raw protein sequence is the most important feature in DeepPPISP. Without the raw protein sequence, ACC, F-measure, and MCC drop from 0.655, 0.397, and 0.206 to 0.564, 0.367, and 0.148, respectively. PSSM and secondary structure are not as important as raw protein sequence. Without PSSM, ACC, F-measure, and MCC drop to 0.605, 0.388, and 0.186, respectively. Without secondary structure, ACC, F-measure, and MCC drop to 0.592, 0.380, and 0.172, respectively. The results indicate that three different types of input features play different roles in DeepPPISP. The most important feature is the raw protein sequence. PSSM and secondary structure are used as auxiliary information to improve the performance of PPI site prediction.

Before we did the experiment, we envisioned that PSSM should be the most useful feature in PPI site prediction because PSSM contains evolutionary information and is the most popular feature. However, the raw protein sequence is the most important feature after a combination of various features. It is of interest to know if the raw protein sequence is also the most important feature in local contextual features. First, we removed the global sequence features, and then compared the performances of different models using only raw protein sequences, PSSM, and secondary structure. From the results presented in Supplementary Table S6, we found that PSSM is the most important feature in local features. By using only PSSM, ACC, F-measure, and MCC drop from 0.520, 0.353, and 0.118 to 0.443, 0.348, and 0.100, respectively. By using only raw protein sequences, ACC, F-measure, and MCC drop to 0.371, 0.339, and 0.07, respectively. By using only secondary structure, ACC, F-measure, and MCC drop to 0.267, 0.334, and 0.05, respectively. Supplementary Fig. S5 shows the ROC curves of DeepPPISP (removing global features) with an individual feature in local contextual features. The AUC value with only PSSM is higher than the others. The results show that the most vital feature is PSSM in local features. This finding is also consistent with previous studies and we believe that the results are reasonable. PSSM contains evolutionary information and has a strong relationship with PPI, and thus almost all sequence-based methods use PSSM as their input features for PPI site prediction.

3.4 The effects of the different lengths of sliding window

Besides the dimension of the global sequence features, we investigated the effects of local contextual features with different sizes. Specifically, we applied different lengths (i.e., 7, 9, 11, 13, and 15) of sliding windows to observe the performance of DeepPPISP. From the results presented in Supplementary Table S7, the best performance is obtained when the length of the sliding window is 7 (the best F-measure and MCC). The overall performances of different lengths of sliding windows are stable. The differences of F-measures and MCCs are very small. The results show that the length of the sliding window can be small if we use global sequence features in our model. We believe that global sequence features already contain some local contextual features, and thus we only need a small sliding window to extract local contextual features.

3.5 The effects of different lengths of proteins

In addition, we believe that the protein length is a very important factor in our study because we need to take global sequence information into consideration. Thus we investigate whether the protein length has an impact on classification results. Protein lengths in our dataset vary from 39 to 869 and 62.1% of the lengths of proteins are less than 200 amino acids. To gain more insights about the effects of protein length, we grouped proteins into short length proteins (less than 200 amino acid residues) and long length proteins (large than 200 amino acid residues). Supplementary Fig. S6 plots the predictive performance of DeepPPISP on the different group. From this figure, the prediction results of short length proteins are consistently higher than the prediction results of long length proteins. The results reveal two phenomena. The first one is that protein length is a very important factor. The results for proteins with different lengths are quite different. The second one is that DeepPPISP is good at predicting short length proteins while not at predicting long length proteins, which is a main limitation of DeepPPISP.

3.6 Case studies

In this section, we give two specific examples of the results obtained by DeepPPISP and other competing methods, to show the real effects of the performance. The two specific examples are not used in training and testing steps. The first example has 14 PPI binding sites (Uniprot ID: P00268). Table 4 lists the positions of correctly predicted PPI sites by DeepPPISP and other competing methods. DeepPPISP correctly predicted 12 PPI sites while PSIVER, SPPIDER, SPRINGS, ISIS, RF_PPI correctly predicted 1, 4, 0, 1, 4 PPI sites, respectively. As can be seen from Table 4, the 12 correctly predicted PPI sites obtained by DeepPPISP contain the correctly predicted PPI sites obtained by other

competing methods. Compared with other methods, DeepPPISP can make a more correct prediction.

Table 4. Position of correctly predicted PPI sites of P00268 by DeepPPISP and other competing methods.

Method	Position of correctly predicted PPI sites
PSIVER	31
SPPIDER	9, 30, 46, 49
SPRINGS	None
ISIS	8
RF_PPI	7, 8, 9, 35
DeepPPISP	4, 5, 7, 8, 9, 30, 31, 35, 46, 47, 49, 50
True binding sites	4, 5, 7, 8, 9, 30, 31, 35, 46, 47, 48, 49, 50, 51

The second example does not have any PPI binding sites (Uniprot ID: P31243). It is worth noting that all proteins in our training and test set have PPI binding sites. Thus it is very interesting to see the results of DeepPPISP on this type of proteins which do not have any PPI binding sites. Table 5 lists the number of predicted PPI sites by DeepPPISP and other competing methods. Using the number as an evaluator, DeepPPISP predicted 23 PPI sites, which is second ranked position on this protein while PEIVER predicted 4 PPI sites. The results show that DeepPPISP does not give a lot of PPI site prediction on proteins which do not have any PPI binding sites.

Table 5. The number of predicted PPI sites of P31243 by DeepPPISP and other competing methods.

Method	# predicted PPI sites
PSIVER	4
SPPIDER	28
SPRINGS	47
ISIS	78
RF_PPI	44
DeepPPISP	23
True binding sites	0

4 Conclusions

Accurate prediction of PPI sites can facilitate the understanding of the biological functions of proteins. In this study, we present a deep learning framework DeepPPISP for the prediction of PPI sites at the residue level. DeepPPISP distinguishes itself from other existing methods is that it combines local and global features which are extracted from protein sequences to predict PPI sites by using deep neural networks. Deep learning techniques have been demonstrated to capture effective features of input data. DeepPPISP uses TextCNN to capture global sequence features, which allows to easily model the relationship between a target amino acid and the whole protein sequence. The results show that DeepPPISP improves PPI site prediction, exceeding existing competing methods. Furthermore, our results demonstrate that the global features of protein sequences can help to improve the prediction of PPI sites. Though DeepPPISP is demonstrated to have advantages over other competing methods, it also has some limitations. The first one is the slow speed. It takes a lot of time to generate sequence profiles (PSSM and DSSP files) and run TextCNN to capture global sequence features of protein sequence in our model. The second one is that DeepPPISP is not good at predicting long length proteins.

Sequence-based PPI site prediction remains a challenging problem. There is no single property from sequence that can analyze protein sequence correctly. In this study, we show that a new feature (global sequence feature) can be used for PPI site prediction. We believe that the global sequence feature has a great potential in other biological sequence analysis and prediction problems. We hope that our study can boost other studies including targeted mutation, drug development and enzymes for various biotechnological applications. In the feature, we would further improve PPI site prediction by enlarging the training set or using more powerful deep learning techniques (Wu, et al., 2019).

Acknowledgements

The authors thank Dr. Kurgan (Virginia Commonwealth University) for constructive discussions. The authors thank anonymous reviewers for valuable suggestions and comments, which greatly improve this paper.

Funding

This work was supported in part by the National Natural Science Foundation of China under Grants (No. 61832019, No. 61622213 and No. 61728211) , the 111 Project (No.B18059), Hunan Provincial Science and Technology Program (2018WK4001), the Fundamental Research Funds for the Central Universities of Central South University (No. 502221903).
Conflict of Interest: none declared

References

Bendell, C.J., *et al.* (2014) Transient protein-protein interface prediction: datasets, features, algorithms, and the RAD-T predictor, *BMC bioinformatics*, **15**, 82.

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 systems biology*, **6**, 128.

Caffrey, D.R., *et al.* (2004) Are protein–protein interfaces more conserved in sequence than the rest of the protein surface?, *Protein Science*, **13**, 190-202.

Carl, N., Konc, J. and Janezic, D. (2008) Protein surface conservation in binding sites, *Journal of chemical information and modeling*, **48**, 1279-1286.

Chen, H. and Zhou, H.X. (2005) Prediction of interface residues in protein–protein complexes by a consensus neural network method: test against NMR data, *Proteins: Structure, Function, and Bioinformatics*, **61**, 21-35.

Choi, Y.S., *et al.* (2009) Evolutionary conservation in multiple faces of protein interaction, *Proteins: Structure, Function, and Bioinformatics*, **77**, 14-25.

De Las Rivas, J. and Fontanillo, C. (2010) Protein–protein interactions essentials: key concepts to building and analyzing interactome networks, *PLoS computational biology*, **6**, e1000807.

de Vries, S.J. and Bonvin, A.M. (2008) How proteins get in touch: interface prediction in the study of biomolecular complexes, *Current protein and peptide science*, **9**, 394-406.

Deng, L., *et al.* (2009) Prediction of protein-protein interaction sites using an ensemble method, *BMC bioinformatics*, **10**, 426.

Fariselli, P., *et al.* (2002) Prediction of protein–protein interaction sites in heterocomplexes with neural networks, *European Journal of Biochemistry*, **269**, 1356-1361.

Guharoy, M. and Chakrabarti, P. (2007) Secondary structure based analysis and classification of biological interfaces: identification of binding motifs in protein–protein interactions, *Bioinformatics*, **23**, 1909-1918.

Han, J.-D.J., *et al.* (2004) Evidence for dynamically organized modularity in the yeast protein–protein interaction network, *Nature*, **430**, 88.

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

Hou, Q., *et al.* (2015) Sequence specificity between interacting and non-interacting homologs identifies interface residues–a homodimer and monomer use case, *BMC bioinformatics*, **16**, 325.

Hou, Q., *et al.* (2016) Club-martini: selecting favourable interactions amongst available candidates, a coarse-grained simulation approach to scoring docking decoys, *PloS one*, **11**, e0155251.

Li, B.-Q., *et al.* (2012) Prediction of protein-protein interaction sites by random forest algorithm with mRMR and IFS, *PloS one*, **7**, e43927.

Li, M.-H., *et al.* (2007) Protein–protein interaction site prediction based on conditional random fields, *Bioinformatics*, **23**, 597-604.

Li, M., *et al.* (2018) Automated ICD-9 Coding via A Deep Learning Approach, *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 1-1.

Li, M., *et al.* (2018) Control principles for complex biological networks, *Briefings in bioinformatics*.

Li, N., Sun, Z. and Jiang, F. (2008) Prediction of protein-protein binding site by using core interface residue and support vector machine, *BMC bioinformatics*, **9**, 553.

Li, X., *et al.* (2019) Network-based methods for predicting essential genes or proteins: a survey, *Briefings in Bioinformatics*.

Li, Z. and Yu, Y. (2016) Protein secondary structure prediction using cascaded convolutional and recurrent neural networks, *arXiv preprint arXiv:1604.07176*.

Lin, X. and Chen, X.w. (2013) Heterogeneous data integration by tree-augmented naïve Bayes for protein-protein interactions prediction, *Proteomics*, **13**, 261-268.

Martin, J. (2014) Benchmarking protein–protein interface predictions: Why you should care about protein size, *Proteins: Structure, Function, and Bioinformatics*, **82**, 1444-1452.

Mihel, J., *et al.* (2008) PSAIA–protein structure and interaction analyzer, *BMC structural biology*, **8**, 21.

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.

Northey, T.C., Barešić, A. and Martin, A.C. (2017) IntPred: a structure-based predictor of protein–protein interaction sites, *Bioinformatics*, **34**, 223-229.

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

Ofran, Y. and Rost, B. (2007) ISIS: interaction sites identified from sequence, *Bioinformatics*, **23**, e13-e16.

Pan, X. and Shen, H.-B. (2018) Predicting RNA–protein binding sites and motifs through combining local and global deep convolutional neural networks, *Bioinformatics*, **34**, 3427-3436.

Porollo, A. and Meller, J. (2007) Prediction-based fingerprints of protein–protein interactions, *Proteins: Structure, Function, and Bioinformatics*, **66**, 630-645.

Russell, R.B. and Aloy, P. (2008) Targeting and tinkering with interaction networks, *Nature chemical biology*, **4**, 666.

Singh, G., *et al.* (2014) SPRINGS: prediction of protein-protein interaction sites using artificial neural networks. PeerJ PrePrints.

Sriwastava, B.K., Basu, S. and Maulik, U. (2015) Protein–protein interaction site prediction in Homo sapiens and E. coli using an interaction-affinity based membership function in fuzzy SVM, *Journal of biosciences*, **40**, 809-818.

Terentiev, A., Moldogazieva, N. and Shaitan, K. (2009) Dynamic proteomics in modeling of the living cell. Protein-protein interactions, *Biochemistry (Moscow)*, **74**, 1586-1607.

Wang, X., *et al.* (2018) Protein–protein interaction sites prediction by ensemble random forests with synthetic minority oversampling technique, *Bioinformatics*.

Wodak, S.J., *et al.* (2013) Protein–protein interaction networks: the puzzling riches, *Current opinion in structural biology*, **23**, 941-953.

Wu, W., Yu, Z. and He, J. (2019) A semi-supervised deep network embedding approach based on the neighborhood structure, *Big Data Mining and Analytics*, **2**, 205-216.

Yan, C., Dobbs, D. and Honavar, V. (2004) A two-stage classifier for identification of protein–protein interface residues, *Bioinformatics*, **20**, i371-i378.

Yaseen, A. and Li, Y. (2013) Dinosolve: a protein disulfide bonding prediction server using context-based features to enhance prediction accuracy, *BMC bioinformatics*, **14**, S9.

Zeng, M., *et al.* (2018) A deep learning framework for identifying essential proteins based on protein-protein interaction network and gene expression data. *2018 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*. IEEE, pp. 583-588.

Zeng, M., *et al.* (2019) A deep learning framework for identifying essential proteins by integrating multiple types of biological information, *IEEE/ACM transactions on computational biology and bioinformatics*.

Zeng, M., *et al.* (2019) Automatic ICD-9 coding via deep transfer learning, *Neurocomputing*, **324**, 43-50.

Zeng, M., *et al.* (2016) Effective prediction of three common diseases by combining SMOTE with Tomek links technique for imbalanced medical data. *2016 IEEE International Conference of Online Analysis and Computing Science (ICOACS)*. IEEE, pp. 225-228.

Zhang, F., *et al.* (2019) DeepFunc: A Deep Learning Framework for Accurate Prediction of Protein Functions from Protein Sequences and Interactions, *Proteomics*, 1900019.

Zhang, J. and Kurgan, L. (2017) Review and comparative assessment of sequence-based predictors of protein-binding residues, *Briefings in bioinformatics*, **19**, 821-837.

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