Predicting Self-Interacting Proteins Using a Recurrent Neural Network and Protein Evolutionary Information

Evolutionary Bioinformatics
Volume 16: 1–9

DOI: 10.1177/1176934320924674
SAGE

Ji-Yong An, Yong Zhou, Zi-Ji Yan and Yu-Jun Zhao

School of Computer Science and Technology, China University of Mining and Technology, Xuzhou, China.

ABSTRACT: Self-interacting proteins (SIPs) play crucial roles in biological activities of organisms. Many high-throughput methods can be used to identify SIPs. However, these methods are both time-consuming and expensive. How to develop effective computational approaches for identifying SIPs is a challenging task. In the article, we present a novel computational method called RRN-SIFT, which combines the recurrent neural network (RNN) with scale invariant feature transform (SIFT) to predict SIPs based on protein evolutionary information. The main advantage of the proposed RNN-SIFT model is that it uses SIFT for extracting key feature by exploring the evolutionary information embedded in Position-Specific Iterated BLAST–constructed position-specific scoring matrix and employs an RNN classifier to perform classification based on extracted features. Extensive experiments show that the RRN-SIFT obtained average accuracy of 94.34% and 97.12% on the *yeast* and *human* dataset, respectively. We also compared our performance with the back propagation neural network (BPNN), the state-of-the-art support vector machine (SVM), and other existing methods. By comparing with experimental results, the performance of RNN-SIFT is significantly better than that of the BPNN, SVM, and other previous methods in the domain. Therefore, we conclude that the proposed RNN-SIFT model is a useful tool for predicting SIPs, as well to solve other bioinformatics tasks. To facilitate widely studies and encourage future proteomics research, a freely available web server called RNN-SIFT-SIPs was developed at http://219.219.62.123:8888/RNNSIFT/ including the source code and the SIP datasets.

KEYWORDS: SIPs, recurrent neural network, scale invariant feature transform, PSSM

RECEIVED: February 14, 2020. **ACCEPTED:** April 16, 2020.

TYPE: Machine Learning Models for Multi-omics Data Integration - Original Research

FUNDING: The author(s) disclosed receipt of the following financial support for the research, authorship, and/or publication of this article: This work is supported by "the Fundamental Research Funds for the Central Universities (2019XKQYMS88)."

DECLARATION OF CONFLICTING INTERESTS: The author(s) declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

CORRESPONDING AUTHOR: Ji-Yong An, School of Computer Science and Technology, China University of Mining and Technology, Xuzhou 21116, Jiangsu, China. Email: ajy@cumt.edu.cn

Introduction

Protein-protein interaction (PPI) prediction revealed multiple roles in many important biological activities. However, an interesting related research problem is whether proteins can interact with their partner. Self-interacting proteins (SIPs) is being considered as a special type of PPIs, which refers to more than 2 copies of the protein can interact with each other and are the same copies of the protein and can be represented by the same gene. This might bring about the formation of homo-oligomer problem. Many recent studies have shown that SIPs play a vital role in various cellular physiological functions and the evolution process of protein-protein interaction networks.[1,2] Therefore, whether a protein can self-interact for interpretation of its functions is very important. The research on SIPs can provide a better understanding of the regulation of protein function and the molecular mechanisms involved in biological activity and the underlying cellular and genetic disease mechanisms. Many studies have been conducted for the homo-oligomerization that is a vital function for biological activity and plays an essential role in a wide range of biological processes, such as signal transduction, gene expression regulation, enzyme activation, and immune response.[3-7] In addition, it has been demonstrated by many previous studies that the diversity function of proteins can be variously extended without increasing the length of genome through SIPs. Self-interacting proteins can also provide some help in improving the protein stability and preventing the protein denaturation by reducing its surface area.[8,9] Therefore, it is becoming more important to develop reliable and effective computational approaches based on protein sequences for predicting SIPs.

Also, more research has been devoted to develop computational methods to predict PPIs. Gao et al[10] proposed a novel computational method called RF-AC, which combined the Rondom Forest (RF) classifier with Autocovariance (AC) approach–based position-specific scoring matrix (PSSM). Huang et al[11] presented a new computational approach, which used weighted sparse representation as classifier and employed global encoding as a feature extraction method for predicting PPIs. Pan et al[12] proposed a novel latent Dirichlet allocation-rondom forest (LDA-RF) model for predicting human PPIs based on protein primary sequences, which has strong ability for processing large-scale datasets by using the LDA-RF model. Zhang[13] proposed a novel approach based on protein sequence that used random tree and genetic algorithm for predicting PPIs. Yang et al[14] presented a new approach that used local descriptors to represent protein sequence and employed the *k*-nearest neighbors for performing classification. Guo et al[15] adopted autocorrelation feature extraction technique for generating feature vectors and used the support vector machine (SVM) classifier to identify PPIs. An et al[16] proposed a classification algorithm of compound kernel function RVM based on gray wolf optimization algorithm and *k*-fold cross-validation, which fully consider the special features of local and global of PPI position. An et al[17] proposed a feature extraction approach based on local protein sequence PSSM matrix coding and serial multifeature fusion. The method can capture

PPI information of continuous and discontinuous for protein sequence by using the local protein sequence PSSM matrix coding; much key feature information–contained protein sequences can be integrated through employing serial multi-feature fusion. These methods usually explore the correlational information between protein pairs, such as coevolution, colocalization, and coexpression. However, this information is not enough for predicting SIPs. In addition, the PPI datasets do not contain the PPIs between the same protein partners. For all these reasons, it is not adequate for predicting SIPs by using these computational approaches. In a previous study, Liu et al[18] proposed a method integrating multiple representative known properties to create a prediction mode called as SLIPPER to predict SIPs. As far as we know, a number of recent studies have been reported about PPIs, which may also be related to SIPs.[19-24] However, there is obviously a drawback that cannot deal with the proteins not covering the current human interatomic by using these methods. Due to all the reasons presented, the development of efficient computational methods for predicting SIPs is a necessary work.

In the study, we proposed a new computational approach called RRN-SIFT, which combines the recurrent neural network (RNN) with scale invariant feature transform (SIFT) for predicting SIPs based on protein evolutionary information. The proposed method uses SIFT to extract key features from PSSM that is constructed by using the Position-Specific Iterated BLAST (PSI-BLAST) tool and contains protein evolutionary information. The RNN classifier is employed for executing classification based on extracted features. The RRN-SIFT model obtained average accuracy of 94.34% and 97.12% on the *yeast* and *human* dataset, respectively. Compared with the back propagation neural network (BPNN), the state-of-the-art SVM, and previous computational models, our method takes full advantage of RNN and SIFT, thereby improving the prediction accuracy. Therefore, the experimental results demonstrated that the proposed RNN-SIFT model is a useful tool for predicting SIPs and is also suitable for other bioinformatics tasks.

Materials and Methods

Dataset

The UniProt database contains 20 199 curated *human* protein sequences.[25] The PPI datasets can be downloaded from different databases, including DIP,[26] BioGRID,[27] IntAct,[28] InnateDB,[29] and MatrixDB.[30] In the article, we constructed the PPI data that only contain the same 2 interaction protein sequences and whose interaction type was defined as "direct interaction" in relevant databases. As a result, we acquired 2994 *human* self-interaction protein sequences. To verify the performance of the RNN-SIFT model, we constructed the experimental datasets by using the following 3 steps: (1) the protein sequences with length less than 50 residues and longer than 5000 residues were removed from the whole human proteome;

$$PSSM = \begin{bmatrix} P_{1,1} & P_{1,2} & P_{1,3} & \cdots & P_{1,20} \\ P_{2,1} & P_{2,2} & P_{2,3} & \cdots & P_{2,20} \\ \vdots & P_{i,j} & \vdots & \vdots & \vdots \\ P_{L,1} & P_{L,2} & P_{L,3} & \cdots & P_{L,20} \end{bmatrix}$$

Figure 1. The schematic of a PSSM.
PSSM indicates position-specific scoring matrix.

(2) we selected the SIP data to create the positive dataset, which must satisfy with 1 of the following conditions: (a) it has been detected for self-interacting by at least 2 kinds of large-scale experiments or 1 small-scale experiment, (b) the protein has been defined as homo-oligomer (including homodimer and homodimers) in UniProt, and (c) it has been reported by at least 2 publications for self-interacting; and (3) for constructing the negative dataset, we removed all types of SIPs from the whole human proteome (including proteins annotated as "direct interaction" and more extensive "physical association") and UniProt database. Consequently, we selected 15 938 non-SIPs as negative samples and 1441 SIPs as positives samples for creating the *human* dataset.[31] In addition, we also used the same strategy to construct the *yeast* dataset that contains 5511 negative and 710 positive samples.[31]

Feature extraction method

Position-specific scoring matrix. Position-specific scoring matrix contains not only the position information but also the evolution information of protein sequence. As a result, the PSSM is used to extract the evolutionary information in the article. Position-Specific Iterated BLAST[32] is used to convert each sequence into a PSSM. Assuming the length of a given protein sequence is L, its PSSM can be expressed as an $L \times 20$ matrix. Figure 1 shows the schematic of a PSSM.

In the artwork, L represents the length of a given sequence, 20 is the number of 20 amino acids, and P_{ij} represents the score of the jth amino acid in the ith position for the query sequence. The P_{ij} is a real value, where if P_{ij} is greater than 0, it means that the ith amino acid is easily mutated into the jth amino acid during the evolution process, and a larger value indicates a higher mutation probability. Conversely, if P_{ij} is less than 0, the position is conservative and the probability of mutation is small. Smaller P_{ij} are more conservative. To extract evolutionary information from protein sequences, each SIP's sequence was converted into a PSSM by using the PSI-BLAST tool. To obtain highly and widely homologous sequences, PSI-BLAST's e-value parameter was set to 0.001 and 3 iterations were selected.

Scale invariant feature transform. Scale invariant feature transform is an image descriptor for image-based matching and recognition developed by Lowe.[33,34] The original SIFT descriptor was calculated from the image intensities around interesting

locations in the image domain which can be named interest points or key points. These interest points are obtained from scale-space extrema of difference of Gaussians (DOG) within a DOG pyramid. Lindeberg[35,36] proposed a new method for finding out interest points by using the SIFT approach. This method can be viewed as a variation of a scale-adaptive blob detection approach, where blobs with associated scale levels are detected from scale-space extrema of the scale-normalized Laplacian. The scale-normalized Laplacian is normalized with respect to the scale level in scale space and is defined as

$$\begin{aligned}\nabla^2_{norm}L\left(x,y^{\cdot},s\right)&=s\left(L_{xx}+L_{yy}\right)\\&=s\left(\frac{\partial^2 L}{\partial x^2}+\frac{\partial^2 L}{\partial y^2}\right)\\&=s\nabla^2\left(G\left(x,y^{\cdot},s\right)*f\left(x,y\right)\right)\end{aligned}\tag{1}$$

For obtaining the maximum value of the DOG image under different scale magnifications, the smoothed image value of a given original image is convolved with Gaussian kernels of different widths by using the SIFT algorithm, a scale-variable Gaussian function is defined as follows

$$G\left(x,y^{\cdot},s\right)=\frac{\left(\frac{1}{2\pi s}e^{-\left(x^2+y^2\right)}\right)}{\left(2s\right)}\tag{2}$$

These Gaussian blurred images are grouped according to their scale magnification, so the number of Gaussian blurred images processed in each group is the same. At this time, the DOG image can be obtained by subtracting 2 adjacent Gaussian blurred images in the same group. The DOG operator constitutes an approximation of the Laplacian operator of different widths, which denotes the standard deviation and the variance of the Gaussian kernel. The DOG operator which constitutes an approximation of the Laplacian operator is defined as follows

$$\begin{aligned}DOG\left(\left(x,y^{\cdot},s\right)&=L\left(x,y^{\cdot},s+\Delta s\right)-L\left(x,y^{\cdot},s\right)\right)\\&\approx\frac{\Delta s}{2}\nabla^2 L\left(x,y^{\cdot},s\right)\end{aligned}\tag{3}$$

Which by the implicit normalization of the DOG responses, as obtained by a self-similar distribution of scale levels $\sigma_{i+1} = k\sigma_i$ used by Lowe, also constitutes an approximation of the scale-normalized Laplacian with $\Delta s\nabla^2 L=\left(k^2-1\right)t\nabla^2 L=\left(k^2-1\right)\nabla^2_{norm}L$ thus implying

$$DOG\left(x,y^{\cdot},s\right)\approx\frac{\left(k^2-1\right)}{2}\nabla^2_{norm}L\left(x,y^{\cdot},s\right)\tag{4}$$

After the DOG image is obtained, the maximum and minimum values can be found and is referred to as key points in the DOG images. To quickly find the key points, each pixel of the DOG image will be compared with 8 pixels around itself and 9 pixels at the same position in the same group of the DOG images at adjacent scales. The maximum and minimum values of these pixels are called key points. As a result, the critical point detection of SIFT algorithm is actually a variant of blob detection, which use Laplacian to compute the maximum value in each magnification space. The Gaussian difference can be approximated as the result of Laplace operator operation. The SIFT employs the concept of "scale space" to capture features at multiple scale levels or image resolutions, which not only increases the number of available features but also makes the method highly tolerant to scale changes.

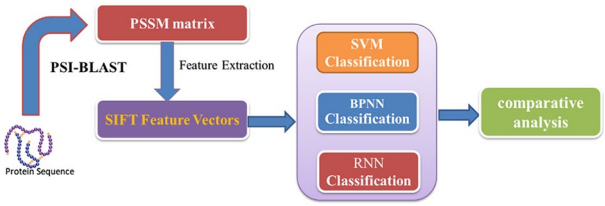

Figure 2. The technology roadmap of the proposed method. BPNN indicates back propagation neural network; PSI-BLAST, Position-Specific Iterated BLAST; PSSM, position-specific scoring matrix; RNN, recurrent neural network; SVM, support vector machine.

In the article, we assumed that each PSSM is an image matrix. As a result, we used the SIFT feature extraction method to generate feature vectors and its dimensional is 128. The technology roadmap of the proposed method is shown in Figure 2.

Recurrent neural network

Recurrent neural network is a machine learning method based on deep learning, which is used to solve binary or multiple classification problems. For tasks that involve sequential inputs, such as speech and language, it is often better to use RNNs. RNNs process an input sequence one element at a time, maintaining in their hidden units a "state vector" that implicitly contains information about the history of all the past elements of the sequence. The final output of the RNN model is the classification label of each feature vector.

Recurrent neural network is used to solve the problem that the input training sample is a continuous sequence and the length of the sequence is different, such as the problem based on time series. The basic neural network only establishes weight connections between layers. The biggest difference of RNN is that the weight connections also established between layers of neurons.[37-39] The structure of RNN is as follows:

It can be seen from Figure 3 that the output of RNN at any moment is related to the current input and the previous output. RNN's forward propagation is a combination of multiplication, addition, and set operations. It is well known that t moment of a given ordered sequence will lead to computation of the hidden layer t times. The current state of hidden layer $h(t)$ is determined by the current input $x(t)$ and the output $h(t-1)$ of the previous layer. The mathematical description is as follows

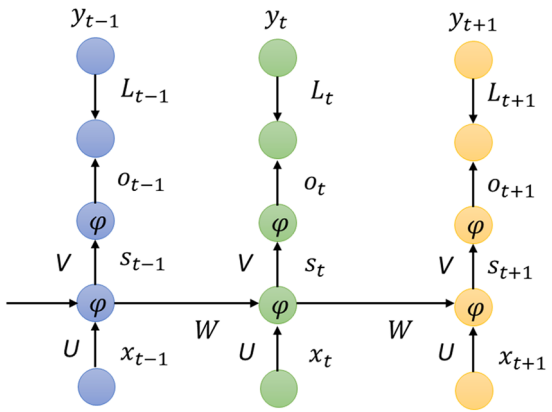

Figure 3. The structure of RNN.
RNN indicates recurrent neural network.

$$s(t) = Ux(t) + Wh(t-1) + b \tag{5}$$

$$h(t) = \sigma(s(t)) = \sigma(Ux(t) + Wh(t-1) + b) \tag{6}$$

where σ represents activation function. The output of the current hidden layer can be calculated by using the following function

$$\sigma(t) = Vh(t) + c \tag{7}$$

The softmax function can be used to perform classification and output the final prediction probability value, which is shown as follow

$$y_p = \sigma(o(t)) = \sigma(Vh(t) + c) \tag{8}$$

Here, the loss function of y_p is different from y. In practice, we can select different loss functions according to the need of the different problem, such as the log loss function, the square loss function, and so on. The loss function of the RNN model at moment t can be expressed as follows

$$Loss_t = -\left[y_t \ln(o_t) + (y_t - 1)\ln(1 - o_t)\right] \tag{9}$$

The loss function (global loss) of the RNN model at all moments N can be expressed as follows

$$Loss = \sum_{t=1}^{N} Loss_t = -\sum_{t=1}^{N}\left[y_t \ln(o_t) + (y_t - 1)\ln(1 - o_t)\right] \tag{10}$$

The gradient of 3 parameters U, V, and W of the global loss can be defined as follows

$$\begin{aligned} \frac{\partial L}{\partial V} &= \sum_{t=1}^{N}(o_t - y_t) \times s_t \\ \frac{\partial L}{\partial U} &= \sum_{t=1}^{N}\sum_{k=1}^{t} \frac{\partial L_t}{\partial s_k^*} \times x_k^T \\ \frac{\partial L}{\partial U} &= \sum_{t=1}^{N}\sum_{k=1}^{t} \frac{\partial L_t}{\partial s_k^*} \times s_{k-1}^T \end{aligned} \tag{11}$$

The most commonly used method for optimization problems is the gradient descent. In the article, the gradient update for the 3 parameters can be expressed as follows

$$\begin{aligned} V &= V - \eta \times \left(\frac{\partial L}{\partial V}\right) \\ U &= U - \eta \times \left(\frac{\partial L}{\partial U}\right) \\ W &= W - \eta \times \left(\frac{\partial L}{\partial W}\right) \end{aligned} \tag{12}$$

The major advantage of the RNN model in learning nonlinear sequential data is well known and has been used in language modeling and sequential labeling. In consideration of SIPs dataset is also a kind of nonlinear sequence data, so we used the RNN model to predict SIPs in the study. The prediction flowchart of RNN-SIFT model is displayed in Figure 4.

Performance evaluation

In the article, we employed the following measures to assess the performance of RNN-SIFT

$$\mathrm{Ac} = \frac{TP + TN}{TP + FP + TN + FN} \tag{13}$$

$$\mathrm{Sn} = \frac{TP}{TP + TN} \tag{14}$$

$$\mathrm{Sp} = \frac{TN}{FP + TN} \tag{15}$$

$$\mathrm{Pe} = \frac{TP}{FP + TP} \tag{16}$$

$$\mathrm{Mcc} = \frac{(TP \times TN) - (FP \times FN)}{\sqrt{(TP + FN) \times (TN + FP) \times (TP + FP) \times (TN + FN)}} \tag{17}$$

where Ac is the accuracy, Sn represents the sensitivity, Sp is the specificity, Pe represents the precision, and Mcc is Matthews's correlation coefficient. *TP* and *TN* represent the number of true interacting and true noninteracting pairs that were correctly predicted, respectively. *FP* and *FN* are the count of true noninteracting pairs and true interacting pairs falsely predicted, respectively. In addition, we used receiver operating curve (ROC) to further evaluate the performance of RNN-SIFT in the experiment.

Results and Discussion

Performance of the proposed RNN-SIFT model

In the experiment, we used the *yeast* and *human* datasets to evaluate performance of the proposed RNN-SIFT model. Generally, overfitting will affect experimental results. Therefore,

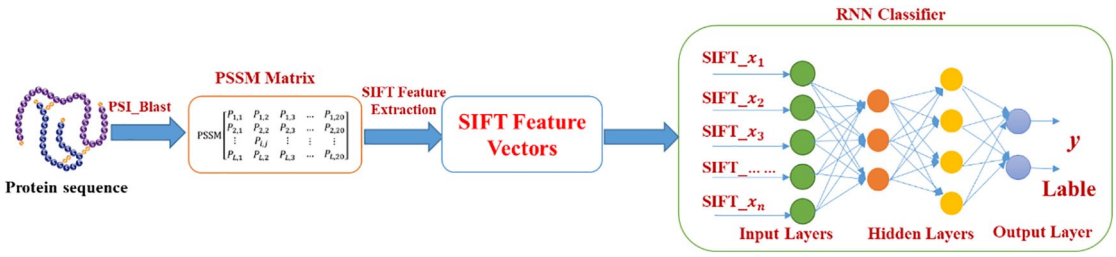

Figure 4. The prediction flowchart of RNN-SIFT.
PSI-BLAST indicates Position-Specific Iterated BLAST, PSSM, position-specific scoring matrix; RNN, recurrent neural network; SIFT, scale invariant feature transform.

Table 1. Fivefold cross-validation results shown using the RNN-SIFT model on *yeast*.

TESTING SET	AC (%)	SN (%)	PE (%)	MCC (%)
1	94.11	64.29	70.79	66.31
2	94.59	70.83	80.19	73.79
3	93.15	62.40	76.47	67.51
4	94.98	65.29	88.76	74.45
5	94.88	72.80	82.73	75.97
Average	94.34 ± 0.74	67.12 ± 4.46	79.79 ± 6.73	71.61 ± 4.38

Abbreviations: Ac, accuracy; Mcc, Matthews's correlation coefficient; Pe, precision; RNN, recurrent neural network; SIFT, scale invariant feature transform; Sn, sensitivity.

Table 2. Fivefold cross-validation results shown using the RNN-SIFT model on *human*.

TESTING SET	AC (%)	SN (%)	PE (%)	MCC (%)
1	97.10	73.27	84.57	77.76
2	97.24	74.89	83.94	80.79
3	96.89	74.59	82.73	79.25
4	96.75	72.45	82.53	77.46
5	97.63	76.37	87.02	81.51
Average	97.12 ± 0.34	83.70 ± 1.15	85.24 ± 2.92	79.35 ±1.79

Abbreviations: Ac, accuracy; Mcc, Matthews's correlation coefficient; Pe, precision; RNN, recurrent neural network; SIFT, scale invariant feature transform; Sn, sensitivity.

we divided the whole datasets into the training datasets and independent test datasets for preventing a biased evaluation. Specifically, we split the *yeast* dataset into 6 parts and selected 5 parts of them as the training set and the remaining dataset selected as independent test dataset. The *human* dataset was also processed by using the same strategy. Meanwhile, 5-fold cross-validation tests were also employed to evaluate prediction ability of the RNN-SIFT for fair comparison, and several parameters of the RNN model were optimized through using the grid search for ensuring fairness. Here, we set up the learning rate = 0.001, training step = 1000, and hidden units = 200. Tables 1 and 2 show the experimental results of the proposed RNN-SIFT model on the *yeast* and *human* datasets.

As can be seen from Table 1, the proposed RNN-SIFT model obtained good experimental results on *yeast* dataset. The result of average accuracy 94.34%, average sensitivity 67.12%, average precision 79.79%, and average Mcc 71.61% was achieved in the experiments on 5-fold cross-validation tests. Similarly, another promising finding from Table 2 was that the RNN-SIFT also achieved better prediction results on *human* dataset, whose average accuracy, sensitivity, precision, and Mcc are 97.12%, 83.70%, 85.24%, and 79.35%, respectively. As a result, the proposed RNN-SIFT model has high value in research.

The good experimental results for predicting SIPs are mainly attributed to use the SIFT feature extraction method and RNN classifier. The main advantage of the RNN-SIFT model is that the SIFT method can extract key evaluation features from PSSM, and the RNN classifier has the advantage of processing sequence data. As discussed, this is mainly due to

Table 3. Fivefold cross-validation results shown by using the BPNN-SIFT model on *yeast*.

TESTING SET	AC (%)	SN (%)	PE (%)	MCC (%)
1	92.86	60.80	75.25	66.11
2	91.70	49.59	70.59	58.06
3	90.64	47.20	65.56	54.81
4	89.86	48.33	57.43	52.31
5	91.47	47.12	61.11	57.84
Average	91.31 ± 1.13	50.61 ± 5.78	65.99 ± 7.15	57.82 ± 5.20

Abbreviations: Ac, accuracy; BPNN, back propagation neural network; Mcc, Matthews's correlation coefficient; Pe, precision; SIFT, scale invariant feature transform; Sn, sensitivity.

Table 4. Fivefold cross-validation results shown by using the SVM-SIFT model on *yeast*.

TESTING SET	AC (%)	SN (%)	PE (%)	MCC (%)
1	89.57	31.63	81.58	49.68
2	90.05	35.33	85.19	55.48
3	89.08	30.40	79.63	48.96
4	90.02	33.88	87.23	52.62
5	89.21	30.12	71.45	46.58
Average	89.58 ± 0.45	33.27 ± 2.26	81.02 ± 6.12	50.66 ± 3.45

Abbreviations: Ac, accuracy; Mcc, Matthews's correlation coefficient; Pe, precision; SIFT, scale invariant feature transform; Sn, sensitivity; SVM, support vector machine.

the following 3 reasons: (1) PSSM contains not only the position information but also the evolution information of protein sequence and retains plenty of prior information. This makes it possible to contain a number of key features that can be extracted. (2) SIFT uses the concept of "scale space" to capture features at multiple scale levels, which not only increases the number of available features but also makes the method highly tolerant to scale changes. This makes it possible for extracting the evolutionary information embedded in PSSM and capturing SIP information. (3) Recurrent neural network has some characteristics in memory, parameter sharing, and Turing completeness, so which provide an advantage for learning based on the nonlinear characteristics of sequences. Therefore, RNN is used to perform classification for predicting SIPs. The results demonstrate 2 things. First, the SIFT method is very suitable for extracting SIP features. Second, the RNN classifier performs well for predicting SIPs, giving good results.

Comparison with the method of BPNN-based and SVM-based

It is interesting to note that the RNN-SIFT model is very suitable for predicting SIPs and can obtain good prediction results. However, to further evaluate the performance of the RNN-SIFT model, we compared the RNN classifier with the BPNN classifier and the SVM classifier by using the same SIFT approach on *yeast* and *human* datasets, respectively. To ensure fair comparison, several parameter settings of BPNN were optimized by employing grid search approach. Specifically, the epochs (the time of training), the eta (learning rate), the BS (the batch size of each training), and the WS (weights) of BPNN are set to 100, 0.006, 0.5, and 0.7. Similarly, by using the same strategy as described above, the RBF kernel parameters of the SVM were optimized, where c is 0.5 and g is 10.8 and other parameters should be take the default values. In addition, the SVM classifier used the LIBSVM tool[40] to perform classification.

Tables 3 to 6 below show the experimental results of BPNN-SIF and SVM-SIFT on the *yeast* and *human* datasets, respectively. Meanwhile, the comparison of ROC curves on the *yeast* and *human* datasets between RNN, BPNN, and SVM is shown in Figures 5 and 6 below, respectively. As outlined in Tables 3 and 4, the BPNN-SIFT model achieved 91.31% average accuracy and the SVM-SIFT model obtained 89.58% average accuracy on *yeast* dataset. Similarly, as can be seen from Tables 5 and 6, the results of average accuracy 93.84% and 91.79% are obtained by the BPNN-SIFT model and the SVM-SIFT model on *human* dataset, respectively. When comparing our results to those of BPNN-SIFT and SVM-SIFT, it

Table 5. Fivefold cross-validation results shown by using the BPNN-SIFT model on *human*.

TESTING SET	AC (%)	SN (%)	PE (%)	MCC (%)
1	94.10	51.61	83.33	68.17
2	94.41	58.30	85.80	73.43
3	93.44	50.41	81.79	66.65
4	92.51	45.28	79.55	62.22
5	94.75	54.85	89.04	68.65
Average	93.84 ± 0.89	52.09 ± 4.89	83.90 ± 3.67	67.82 ± 4.03

Abbreviations: Ac, accuracy; BPNN, back propagation neural network; Mcc, Matthews's correlation coefficient; Pe, precision; SIFT, scale invariant feature transform; Sn, sensitivity.

Table 6. Fivefold cross-validation results shown by using the SVM-SIFT model on *human*.

TESTING SET	AC (%)	SN (%)	PE (%)	MCC (%)
1	92.57	38.21	83.87	57.68
2	91.80	33.33	88.89	52.62
3	90.73	28.00	85.37	47.27
4	91.70	33.88	87.23	51.72
5	92.18	36.00	87.83	56.98
Average	91.79 ± 0.69	33.88 ± 3.81	86.64 ± 2.01	53.23 ± 4.22

Abbreviations: Ac, accuracy; Mcc, Matthews's correlation coefficient; Pe, precision; SIFT, scale invariant feature transform; Sn, sensitivity; SVM, support vector machine.

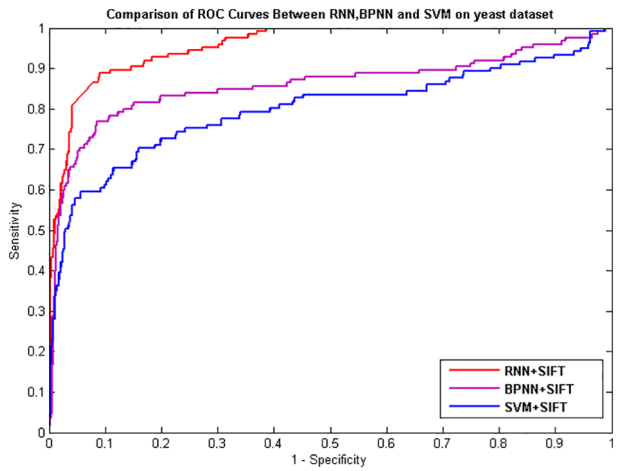

Figure 5. Comparison of ROC curves between RNN, BPNN, and SVM on *yeast* dataset.
BPNN indicates back propagation neural network; RNN, recurrent neural network; ROC, receiver operating curve; SIFT, scale invariant feature transform; SVM, support vector machine.

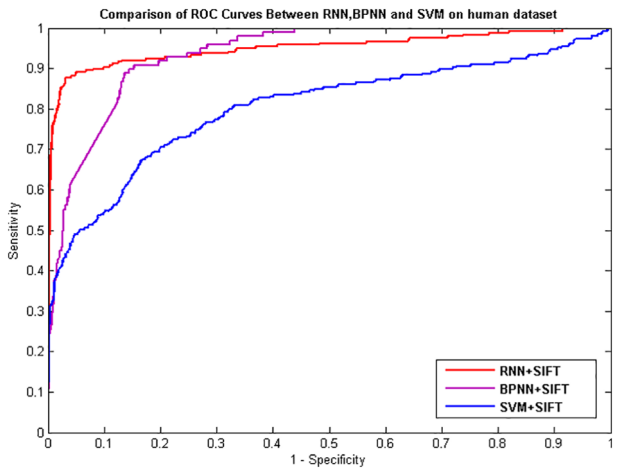

Figure 6. Comparison of ROC curves between RNN, BPNN, and SVM on *human* dataset.
BPNN indicates back propagation neural network; RNN, recurrent neural network; ROC, receiver operating curve; SIFT, scale invariant feature transform; SVM, support vector machine.

must be pointed out that the performance of RNN classifier is significantly better than that of the other 2 classifiers. At the same time, from Figures 5 and 6, the ROC curves of RNN classifier are also significantly better than those of the other 2 classifiers. A major reason for good prediction results is that SIP sequence is nonlinear sequence data, and RNN classifier has some characteristics in memory, parameter sharing, and Turing completeness and can provide an advantage for learning based on the nonlinear characteristics of sequences. From the above analysis, we conclude that the proposed RNN-SIFT model is a useful tool for identifying SIPs, as well as other bioinformatics tasks.

Table 7. Comparison results between RNN-SIFT and other methods on *yeast* dataset.

MODEL	AC (%)	SP (%)	SN (%)	MCC
SLIPPER[18]	71.90	72.18	69.72	0.2842
PPIevo[41]	66.28	87.46	60.14	0.1801
LocFuse[42]	66.66	68.10	55.49	0.1577
CRS[31]	72.69	74.37	59.58	0.2368
SPAR[31]	76.96	80.02	53.24	0.2484
Proposed method	94.34	79.79	67.12	0.7161

Abbreviations: Ac, accuracy; Mcc, Matthews's correlation coefficient; RNN, recurrent neural network; SIFT, scale invariant feature transform; Sn, sensitivity; Sp, specificity.

Table 8. Comparison results between RNN-SIFT and other methods on *human* dataset.

MODEL	AC (%)	SP (%)	SN (%)	MCC
SLIPPER[18]	91.10	95.06	47.26	0.4197
PPIevo[41]	78.04	25.82	87.83	0.2082
LocFuse[42]	80.66	80.50	50.83	0.2026
CRS[31]	91.54	96.72	34.17	0.3633
SPAR[31]	92.09	97.40	33.33	0.3836
Proposed method	97.12	85.24	83.70	0.7935

Abbreviations: Ac, accuracy; Mcc, Matthews's correlation coefficient; RNN, recurrent neural network; SIFT, scale invariant feature transform; Sn, sensitivity; Sp, specificity.

Comparison with other methods

To go a step further and validate the performance of the proposed RNN-SIFT model, we compare the prediction results of the RNN-SIFT model with those of the previous methods, such as SLIPPER,[18]CRS,[31] SPAR,[31] DXECPPI, PPIevo,[41] and LocFuse.[42] Tables 7 and 8 show a detailed comparison results on the *yeast* and *human* datasets. It can be seen from Table 7 that the average accuracy of RNN-SIFT is obviously higher than that of the other 6 approaches on *yeast* dataset. Similarity, Table 8 displays the prediction accuracy obtained by the RNN-SIFT model is also significantly better than that of the other 6 methods on *human* dataset. A similar conclusion was reached by comparing the results from Tables 7 and 8 that the proposed RNN-SIFT model has an excellent prediction capability and can be used for predicting the quality of SIPs. This is a result of using a robust RNN classifier and an effectively SIFT feature extraction technique. These comparison results are further evidence that the RNN-SIFT is suitable for predicting SIPs.

Conclusions

In the study, we proposed a novel computational method called RRN-SIFT, which combines the RNN with SIFT for predicting SIPs based on protein evolutionary information. Extensive experiments show that the RRN-SIFT obtained an average accuracy of 94.34% and 97.12% on the *yeast* and *human* dataset, respectively. We also compared our performance with that of BPNN, the state-of-the-art SVM, and other exiting methods. By comparing with the experimental results, the performance of RNN-SIFT is significantly better than that of the BPNN, SVM, and other previous methods in the domain. This is mainly due to the following 3 reasons: (1) PSSM contains not only the position information but also the evolution information of protein sequence and retains plenty of prior information. This makes it possible to contain a number of key features that can be extracted. (2) Scale invariant feature transform uses the concept of "scale space" to capture features at multiple scale levels, which not only increases the number of available features but also makes the method highly tolerant to scale changes. This makes it possible for extracting the evolutionary information embedded in PSSM and capturing self-protein interaction information. (3) Self-interacting protein sequence is nonlinear sequence data, and RNN has some characteristics in memory, parameter sharing, and Turing completeness and can provide an advantage for learning based on the nonlinear characteristics of sequences. Therefore, we conclude that the proposed RNN-SIFT model is a useful tool for predicting SIPs, as well as to solve other bioinformatics tasks.

Acknowledgments

The authors would like to thank all the guest editors and anonymous reviewers for their constructive advices.

Author Contributions

J-YA conceived the algorithm, performed analyses, prepared the datasets, carried out experiments, wrote the manuscript, and approved the final manuscript.

ORCID iD

Ji-Yong An https://orcid.org/0000-0001-9546-3654

Data Availability

In this study, our experimental datasets contain *yeast* and *human* dataset, which can be obtained from the publicly available DIP,[23] BioGRID,[24] IntAct,[25] InnateDB,[26] and MatrixDB.[27]

REFERENCES

1. Brun VL, Friess W, Schultz-Fademrecht T, Muehlau S, Garidel P. Lysozyme-lysozyme self-interactions as assessed by the osmotic second virial coefficient: impact for physical protein stabilization. *Biotechnol J*. 2010;4:1305-1319.
2. Zhai JX, Cao T-J, An J-Y, Bian Y-T. Highly accurate prediction of protein self-interactions by incorporating the average block and PSSM information into the general PseAAC. *J Theor Biol*. 2017;432:80-86.
3. Baisamy L, Jurisch N, Diviani D. Leucine zipper-mediated homo-oligomerization regulates the Rho-GEF activity of AKAP-Lbc. *J Biol Chem*. 2005;280:15405-15412.
4. Hattori T, Ohoka N, Inoue Y, Hayashi H, Onozaki K. C/EBP family transcription factors are degraded by the proteasome but stabilized by forming dimer. *Oncogene*. 2003;22:1273-1280.
5. Katsamba P, Carroll K, Ahlsen G, et al. Linking molecular affinity and cellular specificity in cadherin-mediated adhesion. *Proc Natl Acad Sci USA*. 2009;106:11594-11599.
6. Koike R, Kidera A, Ota M. Alteration of oligomeric state and domain architecture is essential for functional transformation between transferase and hydrolase with the same scaffold. *Protein Sci*. 2009;18:2060-2066.
7. Woodcock JM, Murphy J, Stomski FC, Berndt MC, Lopez AF. The dimeric versus monomeric status of 14-3-3zeta is controlled by phosphorylation of Ser58 at the dimer interface. *J Biol Chem*. 2003;278:36323-36327.
8. Marianayagam NJ, Sunde M, Matthews JM. The power of two: protein dimerization in biology. *Trends Biochem Sci*. 2004;29:618-625.
9. An JY, Zhang L, Zhou Y, Zhao Y-J, Wang D-F. Computational methods using weighed-extreme learning machine to predict protein self-interactions with protein evolutionary information. *J Cheminform*. 2017;9:47.
10. Gao ZG, Wang L, Xia SX, You ZH, Yan X, Zhou Y. Ens-PPI: a novel ensemble classifier for predicting the interactions of proteins using autocovariance transformation from PSSM. *Biomed Res Int*. 2016;2016:4563524.
11. Huang YA, You Z-H, Chen X, Chan K, Luo X. Sequence-based prediction of protein-protein interactions using weighted sparse representation model combined with global encoding. *BMC Bioinformatics*. 2016;17:184.
12. Pan XY, Zhang YN, Shen HB. Large-scale prediction of human protein-protein interactions from amino acid sequence based on latent topic features. *J Proteome Res*. 2010;9:4992-5001.
13. Zhang L. Sequence-based prediction of protein-protein interactions using random tree and genetic algorithm. Paper presented at: International Conference on Intelligent Computing; July 25-29, 2012; Huangshan, China. https://link.springer.com/chapter/10.1007/978-3-642-31588-6_43.
14. Yang L, Xia JF, Gui J. Prediction of protein-protein interactions from protein sequence using local descriptors. *Protein Pept Lett*. 2010;17:1085-1090.
15. Guo Y, Yu L, Wen Z, Li M. Using support vector machine combined with auto covariance to predict protein–protein interactions from protein sequences. *Nucleic Acids Res*. 2008;36:3025-3030.
16. An JY, Zhou Y, Zhao YJ, Yan ZJ. An efficient feature extraction technique based on local coding PSSM and multifeatures fusion for predicting protein-protein interactions. *Evol Bioinform Online*. 2019;15:1176934319879920.
17. An JY, You ZH, Zhou Y, Wang DF. Sequence-based prediction of protein-protein interactions using gray wolf optimizer-based relevance vector machine. *Evol Bioinform Online*. 2019;15:1176934319844522.
18. Liu Z, Guo F, Zhang J, et al. Proteome-wide prediction of self-interacting proteins based on multiple properties. *Mol Cell Proteomics*. 2013;12:1689-1700.
19. Jia J, Liu Z, Xiao X, Liu B, Chou KC. iPPI-Esml: an ensemble classifier for identifying the interactions of proteins by incorporating their physicochemical properties and wavelet transforms into PseAAC. *J Theor Biol*. 2015;377:47-56.
20. Jia J, Liu Z, Xiao X, Liu B, Chou KC. Identification of protein-protein binding sites by incorporating the physicochemical properties and stationary wavelet transforms into pseudo amino acid composition. *J Biomol Struct Dyn*. 2016;34:1946-1961.
21. Jia J, Liu Z, Xiao X, Liu B, Chou KC. iPPBS-Opt: a sequence-based ensemble classifier for identifying protein-protein binding sites by optimizing imbalanced training datasets. *Molecules*. 2015;21:E95.
22. Hu L, Chan KC. Extracting coevolutionary features from protein sequences for predicting protein-protein interactions. *IEEE/ACM Trans Comput Biol Bioinform*. 2017;14:155-166.
23. Wang Y, You Z-H, Yang S, Li X, Jiang T-H, Zhou X. A high efficient biological language model for predicting protein–protein interactions. *Cells*. 2019;8:122.
24. Kovacs I, Luck K, Spirohn K, et al. Network-based prediction of protein interactions. *Nat Commun*. 2019;10:1240.
25. Consortium UP. UniProt: a hub for protein information. *Nucleic Acids Res*. 2014;43:D204-D212.
26. Xenarios I, Rice DW, Salwinski L, Baron MK, Marcotte EM, Eisenberg D. DIP: the database of interacting proteins. *Nucleic Acids Res*. 2004;32:D449.
27. Stark C, Breitkreutz BJ, Chatr-Aryamontri A, et al. The BioGRID interaction database: 2011 update. *Nucleic Acids Res*. 2011;39: D698-D704.
28. Orchard S, Ammari M, Aranda B, et al. The MIntAct project—IntAct as a common curation platform for 11 molecular interaction databases. *Nucleic Acids Res*. 2014;42:D358-D363.
29. Breuer K, Foroushani AK, Laird MR, et al. InnateDB: systems biology of innate immunity and beyond—recent updates and continuing curation. *Nucleic Acids Res*. 2013;41:D1228-D1233.
30. Launay G, Salza R, Multedo D, Thierrymieg N, Ricardblum S. MatrixDB, the extracellular matrix interaction database: updated content, a new navigator and expanded functionalities. *Nucleic Acids Res*. 2014;43:321-327.
31. Liu X, Yang S, Li C, Zhang Z, Song J. SPAR: a random forest-based predictor for self-interacting proteins with fine-grained domain information. *Amino Acids*. 2016;48:1655-1665.
32. Gribskov M, McLachlan AD, Eisenberg D. Profile analysis: detection of distantly related proteins. *Proc Natl Acad Sci USA*. 1987;84:4355-4358.
33. Lowe DG. Object recognition from local scale-invariant features. Paper presented at: Seventh IEEE International Conference on Computer Vision; September 20-27, 1999; Kerkyra, Greece.
34. Lowe DG. Distinctive image features from scale-invariant keypoints. *Int J Comput Vis*. 2004;60:91-110.
35. Lindeberg T. *Scale-Space Theory in Computer Vision*. Vol. 256. Boston, MA: Springer; 1994:349-382.
36. Lindeberg T. Feature detection with automatic scale selection. *Int J Comput Vis*. 1998;30:79-116.
37. Gregor K, Danihelka I, Graves A, Rezende DJ, Wierstra D. DRAW: a recurrent neural network for image generation. https://arxiv.org/pdf/1502.04623.pdf. Published May 20, 2015.
38. Lukoševičius M, Jaeger H. Reservoir computing approaches to recurrent neural network training. *Comput Sci Rev*. 2009;3:127-149.
39. Donkers T, Loepp B, Ziegler J. Sequential user-based recurrent neural network recommendations. Paper presented at: RecSys'17: Proceedings of the 11th ACM Conference on Recommender Systems; August 27-31, 2017; Como, Italy. https://cseweb.ucsd.edu/classes/fa17/cse291-b/reading/p152-donkers.pdf.
40. Chang C-C, Lin C-J. LIBSVM: a library for support vector machines. *ACM Trans Intell Syst Technol* 2011;2:27.
41. Zahiri J, Yaghoubi O, Mohammad-Noori M, Ebrahimpour R, Masoudi-Nejad A. PPIevo: protein-protein interaction prediction from PSSM based evolutionary information. *Genomics*. 2013;102:237-242.
42. Zahiri J, Mohammad-Noori M, Ebrahimpour R, et al. LocFuse: human protein–protein interaction prediction via classifier fusion using protein localization information. *Genomics*. 2014;104:496-503.