Wang et al. BMC Systems Biology 2018, 12(Suppl 8):129
https://doi.org/10.1186/s12918-018-0647-x

BMC Systems Biology

RESEARCH Open Access

Prediction of protein self-interactions using stacked long short-term memory from protein sequences information

Yan-Bin Wang[1,2†], Zhu-Hong You[1*†], Xiao Li[1*], Tong-Hai Jiang[1], Li Cheng[1] and Zhan-Heng Chen[1,2]

From The International Conference on Intelligent Biology and Medicine (ICIBM) 2018
Los Angeles, CA, USA. 10-12 June 2018

Abstract

Background: Self-interacting Proteins (SIPs) plays a critical role in a series of life function in most living cells. Researches on SIPs are important part of molecular biology. Although numerous SIPs data be provided, traditional experimental methods are labor-intensive, time-consuming and costly and can only yield limited results in real-world needs. Hence,it's urgent to develop an efficient computational SIPs prediction method to fill the gap. Deep learning technologies have proven to produce subversive performance improvements in many areas, but the effectiveness of deep learning methods for SIPs prediction has not been verified.

Results: We developed a deep learning model for predicting SIPs by constructing a Stacked Long Short-Term Memory (SLSTM) neural network that contains "dropout". We extracted features from protein sequences using a novel feature extraction scheme that combined Zernike Moments (ZMs) with Position Specific Weight Matrix (PSWM). The capability of the proposed approach was assessed on *S.erevisiae* and *Human* SIPs datasets. The result indicates that the approach based on deep learning can effectively resist data skew and achieve good accuracies of 95.69 and 97.88%, respectively. To demonstrate the progressiveness of deep learning, we compared the results of the SLSTM-based method and the celebrated Support Vector Machine (SVM) method and several other well-known methods on the same datasets.

Conclusion: The results show that our method is overall superior to any of the other existing state-of-the-art techniques. As far as we know, this study first applies deep learning method to predict SIPs, and practical experimental results reveal its potential in SIPs identification.

Keywords: Self-interacting proteins, Stacked long short-term memory, Deep learning, Dropout

Background

As the embodiment of life activity, protein does not exist in isolation, but through interaction to complete most of the process in the cell. Protein-protein interaction (PPIs) has been the focus of the study of biological processes. SIPs are considered to be a unique protein interaction. SIPs have the same arrangement of amino acids. This leads to the formation of homodimer. Previous studies have proved that SIPs play a leading role in the discovering the laws of life and the evolution of protein interaction networks (PINs) [1]. It is important to understand whether proteins can interact with themselves, which helps clarify the function of proteins, insights into the regulation of protein function, and predicts or prevents disease. The homo-oligomerization have proven to play a significant role in the wide-ranging biological processes, for instance, immunological reaction, signal transduction, activation of enzyme, and regulation of gene expression [2–5]. It has been found that SIPs are a main aspect in regulating protein function by means of allosteric means. Many studies have shown that the

* Correspondence: zhuhongyou@ms.xjb.ac.cn; xiaoli@ms.xjb.ac.cn
†Yan-Bin Wang and Zhu-Hong You contributed equally to this work.
[1]Xinjiang Technical Institute of Physics and Chemistry, Chinese Academy of Science, Urumqi 830011, China
Full list of author information is available at the end of the article

diversity of proteins can be extended by SIPs without growing genome size. In addition, self-interaction helps to increase stability and prevent protein denaturation by reducing its surface area. SIPs have the potential to interact with many other proteins, hence, it occupies a significant position in cellular systems. SIPs have an ability to improve the stability of protein and avoid the denaturation of proteins and reduce its superficial area. An endless stream of experimental methods is used to detect protein self-interaction. However, these methods have certain drawbacks and limitations. It is urgent to develop an effective and reliable novel approach for predicting SIPs.

In recent years, some computational systems have been designed for predicting PPIs. Zaki et al. [6] projected a scheme for predicting SIPs that used only protein primary structure based on pairwise similarity theory. Zahiri J at el. [7] introduced an approach called PPIevo for predicting PPIs using a feature extraction algorithm. You et al. [8] gave a method called PCA-ELM that shows great ability in predicting PPIs. M. G. Shi et al. [9] shown a powerful method, which used correlation coefficient (CC) combined with support vector machine (SVM). This proposed method could be used in predicting PPIs, giving satisfactory results. These methods generally tend to use certain information about protein pairs, for instance, colocalization, coexpression and coevolution. Nevertheless, such feature is not applicable to deal with SIPs problems. Besides, the PPIs data sets adopted in above approaches do not cover SIPs. Hence, these computational-based methods not suitable for predicting SIPs. In the past research, Liu et al. [10] developed a prediction model to predict SIPs named as SLIPPER by mixing several typical known attributes. However, there is a major defect in this prediction model, which cannot deal with proteins that are not included in the current human interatomic. Given the limits of the above-mentioned approaches, it is needed to develop a more practical computational method for identifying SIPs.

In this study, a novel computational scheme based on deep learning named ZM-SLSTM is proposed for detecting SIPs from protein sequence. We firstly converted the SIPs sequence into Position Specific Weight Matrix (PSWM). Second, a novel feature extraction approach named as Zernike moments (ZMs) is adopted to generate feature vector from PSWM. Then, we build a Stacked Long Short-Term Memory (SLSTM) to predict SIPs. The proposed model was executed on *S.erevisiae* and *human* SIPs data sets. Satisfactory results are obtained with high accuracy of 95.69 and 97.88%, respectively. This method is also compared with other methods including Support Vector Machine (SVM), other (named as SLIPPER, CRS, SPAR DXECPPI, PPIevo and LocFuse). The results show that the ZM-SLSTM method perform better than any those methods. For all we know, our study is the first to adopt the deep-learning technology to predict SIPs, and experimental results show that our method can effectively resist data skew and improve the prediction performance relative to the existing technique.

Method

Datasets

We download 20,199 data of human sequences protein from the Uniprot database [11]. The PPIs data come from Various resource libraries including MatrixDB, BioGRID, DIP, IntAct and InnateDB [12–16]. In order to obtain the SIP data set, the PPI data that can interact with itself were collected. Accordingly, we obtained 2, 994 human SIPs sequences.

To collect datasets scientifically and efficiently, the human SIPs dataset is screened by the following steps [17]: (1) the protein sequence(>5000residues or < 50 residues) was removed from the whole human sequences protein; (2) For the construction of the positive data set, the selected SIPs must meet one of the following situations: (a) At least two mass experiments or one small scale experiment have shown that this protein sequence can interact with itself; (b) the protein must be homooligomer in UniProt; (c) the self-interaction of this protein have been reported by more than one publication; (3) For the sake of establish negative data set, all known SIPs were deleted from the whole human proteome.

As a result, 1441 human SIPs were selected to build positive data sets and 15,938 human protein that non-interacting were selected to build negative datasets. In addition, to better verify the usefulness of the designed scheme, we constructed the *S.erevisiae* SIPs dataset that cover 710 SIPs and 5511 non-SIPs by using above strategy.

Position specific weight matrix

PSWM [18] was first adopted for detecting proteins of distantly related. The PSWM successfully applied in the field of biological information, including protein disulfide connectivity, protein structural classes, and subnuclear localization, DNA or RNA binding sites [19–23]. In the study, we used PSWM for predicting SIPs. A PSWM for a query protein is a Y×20 matrix $M = \{m_{ij}: i = 1 \cdots Y \text{ and } j = 1 \cdots 20\}$, where the Y represents the size of the protein sequence and the number of columns of M matrix denotes 20 amino acids. In order to construct PSWM, a position frequency matrix is first created by calculating the presence of each nucleotide on each position. This frequency matrix can be represented as $p(u, k)$, where u means position, k is the k_{th} nucleotide. The PSWM can be expressed as $M_{ij} = \sum_{k=1}^{20} p(u,k) \times w(v,k$

), where $w(v, k)$ is a matrix whose elements represent the mutation value between two different amino acids. Consequently, high scores represent highly conservative positions, and low points represent a weak conservative position.

In this paper, the PSWM of a protein sequences were generated by using Position specific iterated BLAST (PSI-BLAST) [24]. To get high and broad homologous information, we set three iterations and set the e-value to 0.001.

Zernike moments

In this paper, the Zernike moments are introduced to extract meaningful information from protein sequence and generate feature vector [25–30]. We introduce the concept of the Zernike function to clearly define the moments of the Zernike. A set of complex polynomials are introduced by Zernike which form a complete orthogonal set within the unit circle. These polynomials are represented as $V_{nm}(x, y)$. These polynomials have the following form:

$$V_{xy}(n, m) = V_{xy}(\rho, \theta) = R_{xy}(\rho)e^{jy\theta} \text{for } \rho \le 1 \quad (1)$$

where x is a positive integer greater than zero, y is integer, and satisfies $|y| < x$, where x - $|y|$ is an even number. ρ is the length from (0, 0) to the pixel (n, m). θ represents included angle between vector ρ and n axis in counterclockwise direction. $R_{xy}(\rho)$is

$$R_{xy}(\rho) = \sum_{s=0}^{(x-|y|/2)} (-1)^s \frac{(x-s)!}{s!\left(\frac{x+|y|}{2}-s\right)!\left(\frac{x+|y|}{2}-s\right)!} \rho^{x-2s} \quad (2)$$

From equation (2), we can find $R_{x,-y}(\rho) = R_{xy}(\rho)$. These orthogonal polynomials are satisfying:

$$\int_0^{2\pi}\int_0^1 V_{xy}^*(\rho, \theta) V_{pq}(\rho, \theta)\rho d\rho d\theta = \frac{\pi}{x+1}\delta_{xp}\delta_{yq} \quad (3)$$

with

$$\delta_{ab} = \begin{cases} 1 & a = b \\ 0 & otherwise \end{cases} \quad (4)$$

The Zernike moments can be obtained by calculating (5)

$$Z_{xy} = \frac{x+1}{\pi}\sum_{(\rho,\theta)\in unit\ circle}\sum f(\rho, \theta)V_{nm}^*(\rho, \theta) \quad (5)$$

To calculate the ZMs of a protein sequence represented by a PSWM matrix, the origin is at the center of the matrix, and the points in the matrix are mapped inside the unit circle., i.e., $n^2 + m^2 \le 1$. The value falling outside the unit circle is not calculated [31–35]. Note that $A_{xy}^* = A_{x,-y}$.

Feature selection

To sum up, Zernike moments can extract some important information. When we use the Zernike moments, there is a problem that must be considered is how big n_{max}should be set? The moments of lower order extract unsophisticated feature and the moments of higher order capture details feature. Figure 1 shows the magnitude plots of the Zernike moments with low order. Considering that we not only need enough information for more accurate classification, but also need to control the dimension of feature to reduce the computational cost. In this experiment, x_{max} is set to 30 [36–40]. This moment information constitutes the feature vectors of protein sequences

$$\overrightarrow{F} = [|A_{11}|, |A_{22}|, ..., |A_{NM}|]^T \quad (6)$$

where $|A_{nm}|$ represents the absolute value of Zernike moments. The zeroth order moments are not computed because they do not contain any valuable information and ZMs without considering $m < 0$, since they are inferred through $A_{n,-m} = A_{nm}^*$.

Finally, in order to eliminate noise as much as possible and to reduce the computational complexity, the feature dimensional was reduced from 240 to 150 by means of principal component analysis (PCA) method [41].

Long short-term memory

Long Short-Term Memory (LSTM), a special recurrent neural network, performs much better than standard recurrent neural networks in many tasks. Almost all exciting results based on recurrent neural networks are implemented by them. In this work, the deep LSTM net structure was first introduced to predict self-interaction protein.

The main difference between LSTM network and other networks is its use of complex memory block instead of the neurons of general network. The memory block contains three multiplicative 'gate' units (the input, forget, and output gates.) along with some memory cells (one or more). The gate unit is used to control the information flow, and the memory cell is used to store the historical information [42–44]. The structure of the memory block is shown in the Fig. 2, to better understand the work of the gate unit, memory cells are not shown in the Fig. 2. The gate removes or restore information to the cell state by controlling the information flow. More specific, the input and output of the information flow are respectively handled by the input and output gates. The forget gate determines how much of the

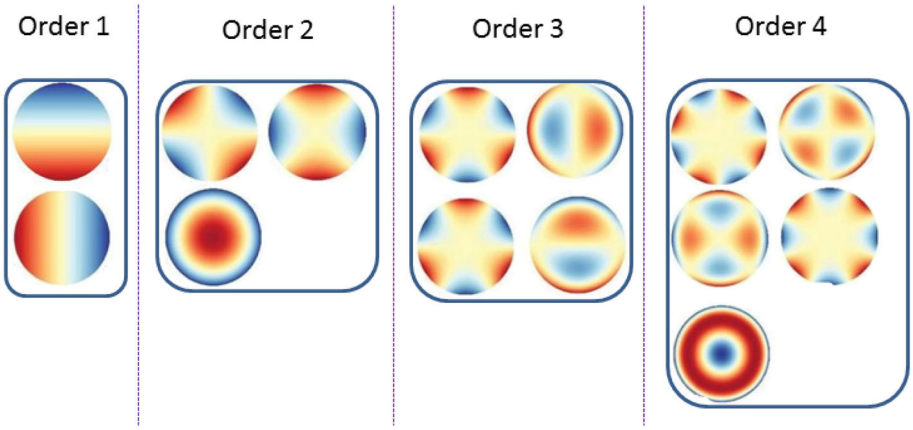

Fig. 1 Plots of the magnitude of the Zernike moments with low order

previous unit's information is retained to the current unit. In addition, in order to enable memory blocks to store earlier information, we add a peephole to the block to connect the memory cell to the gate [45, 46].

The information flow passing through a memory block needs to do the following operations to complete the mapping from input x to output h:

$$i_t = sigm(W_i \cdot [C_{t-1}, x_t, h_{t-1}] + b_i) \tag{7}$$

$$f_t = sigm(W_f \cdot [C_{t-1}, x_t, h_{t-1}] + b_f) \tag{8}$$

$$o_t = sigm(W_o \cdot [C_t, x_t, h_{t-1}] + b_o) \tag{9}$$

$$\check{C}_t = tanh(W_C \cdot [x_t, h_{t-1}] + b_C) \tag{10}$$

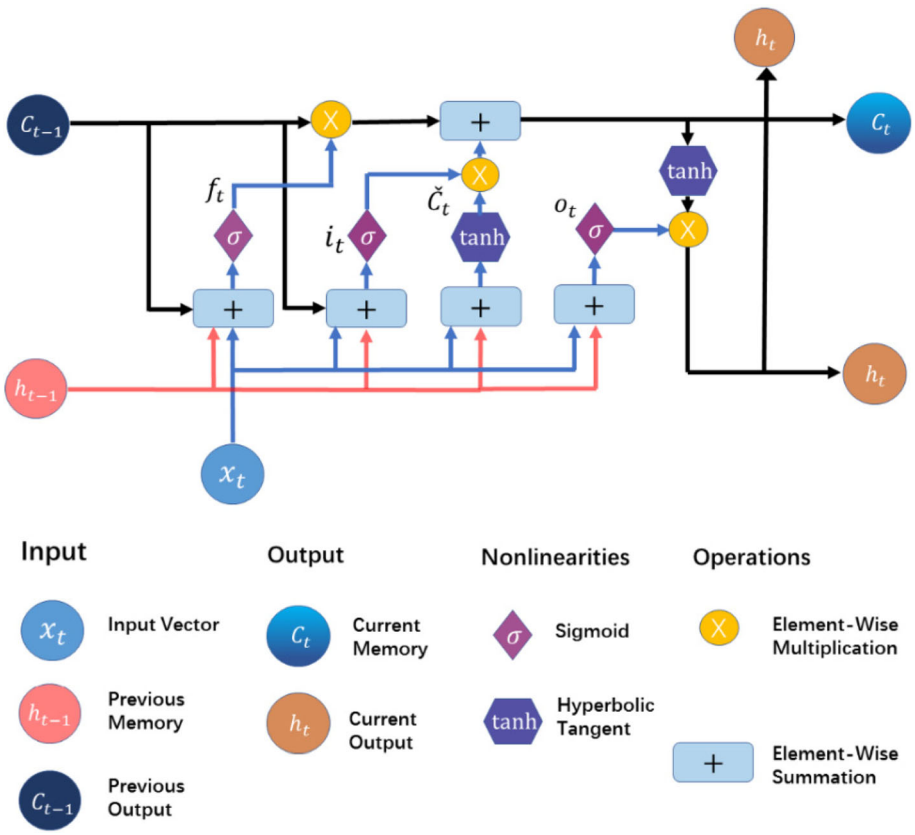

Fig. 2 The structure of memory blocks in SLSTM networks

$$C_t = C_{t-1} * f_t + \tilde{C}_t * i_t \tag{11}$$

$$h_t = \tanh(C_t) * o_t + \tilde{C}_t * i_t \tag{12}$$

Here, symbols related to the letter C represent cell activation vectors, the symbol f, i, o, and C are respectively the forget gate, input gate, output gate. The items related to W (W_i, W_f, W_f), represent weight matrices, the items related to b (b_i, b_f, b_o, b_C) denote bias, σ is sigmoid function, $*$ is the element-wise product of the vectors.

Stacked long short-term memory

A large number of theoretical and practical results support that the deep hierarchical network model can be more competent for complex tasks than shallow one. We construct the Stacked Long Short-Term Memory (SLSTM) net by stacking multiple LSTM hidden layers on top of each other, which contain one input layer, three LSTM hidden layers, one output layer. Figure 3 shows a SLSTM network. The number of neurons in the input layer is equal to the dimension of the input data. Each SLSTM hidden layer consist of 16 memory blocks. The number of neurons in the output layer equals the number of classes. Therefore, the number of neurons or memory blocks in each layer of the network are *200–16–16-16-2*. In output layer, the softmax function is used to generate probabilistic results.

Prevent over fitting

Overfitting problems exist in many prediction or classification models. Even the deep learning model with superior performance is no exception. A great deal of theoretical and practical work has proved that over-fitting can be reduced or avoided by adding "dropout" operation on neural net. "dropout" provides a way to approximate combine exponentially different neural network architectures [47]. More specific, "dropout" involves two important operations: 1) Dropout randomly discards hidden units and edges connected with them with a fixed probability in each training case; 2) In the test, dropout is responsible for integrating multiple neural networks generated during training. The first operation makes it possible to produce a different network almost every training case and these different networks share the same weights for the hidden units. The Fig. 4 describes a network model after using dropout. At test time, all hidden layer neurons are used without "dropout", but the weight of the network is a reduced version of the trained weights. The proportion of weight reduction equals to the probability of the unit being retained [48]. By weight reduction, a large number of dropout networks can be merged into a single neural network and provide a similar performance to averaging over all networks [49].

Results

Performance evaluation

In order to evaluate the methods presented in this paper, we used a few commonly used indicators: The accuracy (ACC), true positive rate (TPR), positive predictive value (PPV), specificity (SPC), and Matthew's Correlation Coefficient (MCC). The definition is given as follows:

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

$$\mathrm{TPR} = \frac{TP}{FN + TP} \tag{14}$$

$$\mathrm{PPV} = \frac{TP}{TP + FP} \tag{15}$$

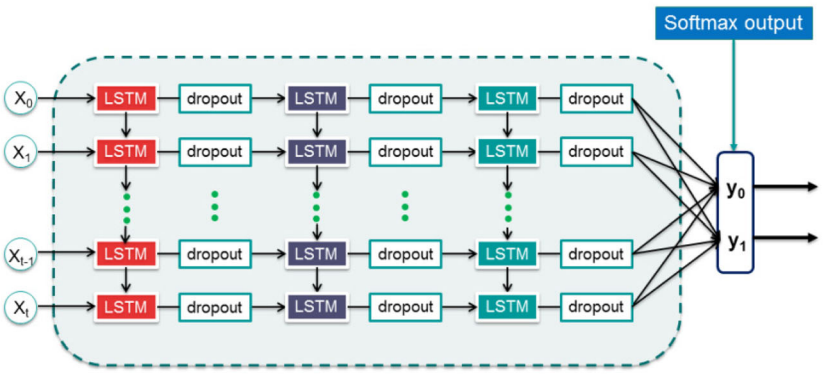

Fig. 3 A Stacked Long Short-Term Memory network

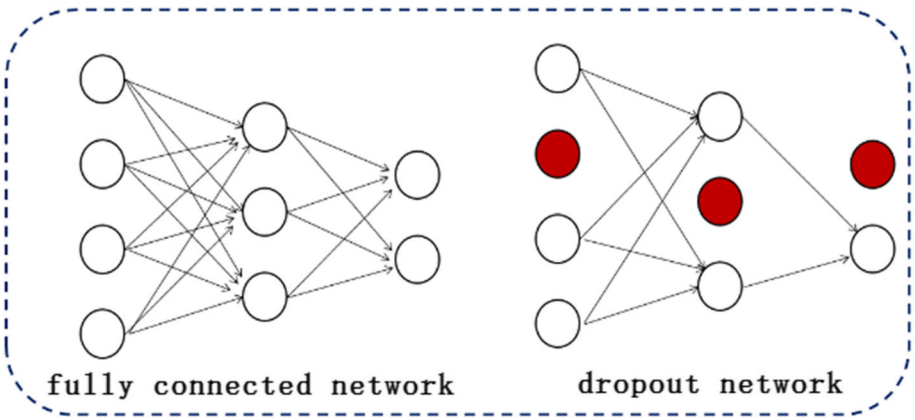

Fig. 4 Network structure after using dropout

$$\mathrm{SPC} = \frac{TN}{TN + FP} \tag{16}$$

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

where *TP* means those samples, have interacting, are predicted correctly, *FP* represents those samples, true non-interacting with each other, are judged to be interaction. *TN* represents those samples, true noninteracting with each other, are predicted correctly. *FN* represents those samples, true interacting with each other, are judged to be non-interacting. Furthermore, the Receiver operating characteristic (ROC) is portrayed to appraise the performance of a set of classification results and the AUC is computed as an important evaluation indicator [50, 51].

Assessment of prediction

The proposed method is validated on two standard SIPs dataset. Each dataset is divide into three parts: The training set, accounted for 40 % of the total data; The verification set, accounts for 30 % of the total data; and the test set, accounts for 30 % of the total data. The training data sets are used to fit the weights of connections between memory block in the SLSTM network. The validation sets are used to fine tune model parameters and determine optimal performance models. Another function of the validation data set is to prevent overfitting by early stopping: when the errors on the validation data set begin to increase, the model stops training, because is a token of overfitting. The test data set is used for unbiased evaluation of the trained model. We train model only setting 200 epochs and using *Nadam* optimization method, that has more constraints on the learning rate, and also has a more direct impact on the gradient update.

As Table 1 shows, the accuracy obtained by the ZMs-SLSTM is 95.69% for *S.erevisiae* and 97.88% for *Human* data sets. Beyond that, several other evaluation indicators also show the potential of our approach. More specifically, on *S.erevisiae,* the proposed method achieved TPR of 92.97%, SPC of 95.94%, PPV of 67.23%, MCC of 77.43% and AUC of 0.9828, respectively. For *Human* dataset with more samples, this method produces better results with TPR of 88.00%, SPC of 98.70%, PPV of 84.93%, MCC of 85.60% and AUC of 0.9908, respectively. The ROC curves achieved by the proposed ZMs-SLSTM method was exposed in Fig. 5.

The performance of SVM-based approach

We verify the performance of our classifier by compare it with the SVM (Support Vector Machine) classifier representing the most advanced technologies. In this experience, we took the same feature extraction process in *S.erevisiae* and *Human* datasets, respectively. We used LIBSVM tools [52] to implement the classification of SVM. The SVM parameters of c and g are 0.5 *and* 0.6 by the grid search method.

Table 1 The results produced by the proposed method and the SVM-based method on PPIs datasets

Model	Data Sets	ACC (%)	TPR (%)	SPC (%)	PPV (%)	MCC (%)	AUC
SLSTM	*S.erevisiae*	95.69	92.97	95.94	67.23	77.43	0.9828
	Human	97.88	88.00	98.70	84.93	85.60	0.9908
SVM	*S.erevisiae*	93.06	57.22	97.68	76.25	64.59	0.9345
	Human	95.30	54.26	99.01	83.27	66.07	0.9261

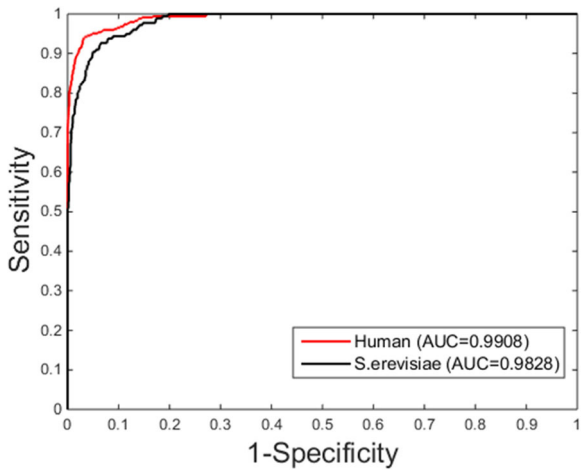

Fig. 5 ROC curves achieved by the proposed approach

Table 1 indicates, our ZMs-SLSTM method is significantly superior to SVM-based methods, particularly for predicting the true self-interacting protein pairs. Focus on *S.erevisiae* dataset, 95.69% ACC, 92.97% TPR, 77.43% MCC and 0.9828 AUC of the ZMs-SLSTM is much higher than the corresponding values for the SVM-predictor with 93.06% ACC, 57.22%TPR, 64.59% MCC and 0.9345. AUC. Similar situations also appear on the *Human* data set, the performance of the ZMs-SLSTM method has been found to be better with 97.88% ACC, 88.00% TPR, 98.70% SPC, 84.93% PPV, 85.60% MCC and 0.9908 AUC versus 95.30% ACC, 54.26% TPR, 99.01% SPC, 83.27% PPV, 66.07% MCC and 0.9261 AUC, respectively. In particular, higher TPR (92.97% on *S.erevisiae* dataset and 88.00% on *Human* dataset) indicates our method can give more accurate results than SVM-based approach (57.22% on *S.erevisiae* dataset and 54.26% on *Human* dataset) in predicting true SIPs.

Comparison with other methods

To further evaluate our proposed approach, we also compared it with six existing methods (SLIPPER, CRS, SPAR, DXECPPI, PPIevo and LocFuse). Table 2 presents the results of several methods on *S.erevisiae* and *Human* data sets. From Table 2, compared with other methods, our method significantly improves the overall performance of the SIPs prediction. In addition, SLIPPER contains some restrictions. Second, it integrates a large amount of known knowledge, such as GO terms, PINs, drug targets, and enzymes. In particular, the degree of protein in the PIN makes a significant contribution to SIP predictions. However, for unknown or artificial proteins in actual applications, all information is difficult to access directly. Therefore, as long as the protein sequence is known, our method is necessary for improved SIP prediction. DXECPPI is a PPI predictor, because the traditional PPI predictor uses correlation information between two proteins, such as co-expression, co-evolution and co-localization, and cannot be effectively used for SIP prediction. Therefore, our method can be used as a necessary supplement for PPI prediction. For *S.erevisiae* data set, the method presented by this paper achieves the best accuracy of 95.69%, which is much higher than that of other methods. More obvious improvements are reflected in TPR, MCC, and AUC. Observe the results on the *S.erevisiae* data set, 92% TPR achieved by the ZMs-SLSTM approach is more than three triple that of the DXECPPI method, and 77.43% MCC achieved by the ZMs-SLSTM approach is more than four triple that of the PPIevo method. 0.9828 AUC achieved by the ZMs-SLSTM approach is 37% higher than the average of other methods. High TPR shows that our method has little error rate in identifying self-interacting proteins. The high MCC and AUC show that our model is robust, practical, and can effectively resist data skew. The SIP prediction for *Human* dataset (Table 2) have also been greatly improved by using our approach. 97.88% ACC, 85.60% MCC and 0.9908 AUC of the ZMs-SLSTM is are way above the corresponding values for the other method. In addition, compared the results of SVM-based method (Table 1) and six existing methods (SLIPPER, CRS, SPAR, DXECPPI, PPIevo and LocFuse), it can be found that our method is still overall superior to the six existing predictors. This shows that

Table 2 Performance comparison of seven approaches on both the *S.erevisiae* and *Human* datasets

Methods	*S.erevisiae*					*Human*				
	ACC (%)	SPC (%)	TPR (%)	MCC (%)	AUC	ACC (%)	SPC (%)	TPR (%)	MCC (%)	AUC
SLIPPER	71.90	72.18	69.72	28.42	0.7723	91.10	95.06	47.26	41.97	0.8723
DXECPPI	87.46	94.93	29.44	28.25	0.6934	30.90	25.83	87.08	8.25	0.5806
PPIevo	66.28	87.46	60.14	18.01	0.6728	78.04	25.82	87.83	20.82	0.7329
LocFuse	66.66	68.10	55.49	15.77	0.7087	80.66	80.50	50.83	20.26	0.7087
CRS	72.69	74.37	59.58	23.68	0.7115	91.54	96.72	34.17	36.33	0.8196
SPAR	76.96	80.02	53.24	24.84	0.7455	92.09	97.40	33.33	38.36	0.8229
ZM-SLSTM	95.69	95.94	92.97	77.43	0.9828	97.88	98.70	88.00	85.60	0.9908

the proposed feature extraction strategy proposed in this paper is efficient, useful and plays an important role in the SIPs prediction model. The results of this study illustration that the ZMs-SLSTM approach is capable of effectively improving the prediction performance of SIPs.

Discussion

This method can produce good results mainly due to: effective feature extraction strategy and reliable classifiers.

The protein feature extraction scheme consisting of PSWM and ZMs effectively captures the evolutionary information of protein and produces the most characteristic features that improve the ability of the classifier to distinguish unknown samples during the testing phase. The robust and efficient SLSTM deep neural network also make a great contribution to accuracy improvement that provide stronger classification performance than traditional machine learning method in interaction pattern recognition. The performance improvement brought by SLSTM comes mainly from the following reasons: 1) Compared with the traditional machine learning methods, the hierarchical structure of deep learning algorithms can process more complex data, and automatically learn abstract and more useful features. 2) Two mechanisms to prevent overfitting, dropout and early stopping, make the prediction model trained more reliable, robust and excellent. 3) In the testing phase, we merged all dropout networks generated by the training processes, which led to a better result. 4) The SLSTM network uses memory blocks instead of simple neurons, which allows the network to learn more knowledge about self-interacting proteins during training.

Conclusion

In recent years, the rise of deep learning technology has constantly affected the development of various fields. However, the ability of deep learning techniques in predicting self-interacting proteins has not been witnessed. In this work, a SLSTM neural network was constructed as a deep learning model to predict SIPs only using protein sequences. The method is applied to two standard data sets and the results show it is reliable, stable and accurate for predicting SIPs. The contribution of the proposed approach comes mainly from three technologies: SLSTM network, ZMs feature extractor, PSWM. Specifically, each protein sequence was converted into PSWM by using PSI-BLAST. The ZMs then is adopted to catch the valuable information from PSWM and form feature vectors that as input of classifier. Finally, the SLSTM deep network is used to predict SIPs. For further measuring the performance of the ZMs-SLSTM method, ZMs-SVM and other six methods were implemented on *S.erevisiae* and *Huamn* data sets for comparing with the proposed approach. The results from these experiments indicate that the SIPs detection capability of the proposed scheme is overall ahead of that of the earlier methods and SVM-based approach. The performance improvement caused by this method is mainly dependent on the use of an excellently deep learning model and a fresh and high-performance feature extraction scheme. To the best of our knowledge, this study is the first to build a deep learning model for SIP prediction using protein sequence, and the results demonstrate our method is strong and practical.

Abbreviations

ACC: Accuracy; MCC: Matthew's correlation coefficient; PINs: Protein interaction networks; PPIs: Protein-protein interaction; PPV: Positive predictive value; PSWM: Position specific weight matrix; SIPs: Self-interacting proteins; SLSTM: Stacked long short-term memory; SPC: specificity; SVM: Support vector machine; TPR: True positive rate; ZMs: Zernike moments

Acknowledgments

Not applicable.

Funding

Publication of this article was sponsored in part by the National Science Foundation of China, under Grants 61722212 and 61572506, in part by the Pioneer Hundred Talents Program of Chinese Academy of Sciences. The authors would like to thank all anonymous reviewers for their highly-developed advices.

Availability of data and materials

https://figshare.com/s/0d99da1a33850136e2cf

About this supplement

This article has been published as part of *BMC Systems Biology Volume 12 Supplement 8, 2018: Selected articles from the International Conference on Intelligent Biology and Medicine (ICIBM) 2018: systems biology*. The full contents of the supplement are available online at https://bmcsystbiol.biomedcentral.com/articles/supplements/volume-12-supplement-8.

Authors' contributions

YBW and ZHY considered the algorithm, carried out analyses, arranged the data sets, carried out experiments, and wrote the manuscript. XL, THJ, LC and ZHC designed, performed and analyzed experiments. All authors read and approved the final manuscript.

Ethics approval and consent to participate

Not Applicable.

Consent for publication

Not Applicable.

Competing interests

The authors declare no conflicts of interest.

Publisher's Note

Author details

[1]Xinjiang Technical Institute of Physics and Chemistry, Chinese Academy of Science, Urumqi 830011, China. [2]University of Chinese Academy of Sciences, Beijing 100049, China.

Published: 21 December 2018

References

1. Ispolatov I, Yuryev A, Mazo I, Maslov S. Binding properties and evolution of homodimers in protein–protein interaction networks. Nucleic Acids Res. 2005;33(11):3629–35.

2. Park HK, Lee JE, Lim J, Jo DE, Park SA, Suh PG, Kang BH. Combination treatment with doxorubicin and gamitrinib synergistically augments anticancer activity through enhanced activation of Bim. BMC Cancer. 2014;14(1):431.
3. Katsamba P, Carroll K, Ahlsen G, Bahna F, Vendome J, Posy S, Rajebhosale M, Price S, Jessell TM, Ben-Shaul A. Linking molecular affinity and cellular specificity in cadherin-mediated adhesion. Proc Natl Acad Sci. 2009;106(28):11594.
4. Baisamy L, Jurisch N, Diviani D. Leucine zipper-mediated homo-oligomerization regulates the rho-GEF activity of AKAP-Lbc. J Biol Chem. 2005;280(15):15405–12.
5. 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(10):2060–6.
6. Nazar Z, Sanja LM, Wassim EH, Piers C. Protein-protein interaction based on pairwise similarity. BMC Bioinformatics. 2009;10(1):1–12.
7. 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(4):237–42.
8. You ZH, Lei YK, Zhu L, Xia J, Wang B. Prediction of protein-protein interactions from amino acid sequences with ensemble extreme learning machines and principal component analysis. BMC Bioinformatics. 2013;14(8):1–11.
9. Shi MG, Xia JF, Li XL, Huang D. Predicting protein–protein interactions from sequence using correlation coefficient and high-quality interaction dataset. Amino Acids. 2010;38(3):891.
10. Liu Z, Guo F, Zhang J, Wang J, Lu L, Li D, He F. Proteome-wide prediction of self-interacting proteins based on multiple properties. Mol Cell Proteomics. 2013;12(6):1689.
11. Consortium UP. UniProt: a hub for protein information. Nucleic Acids Res. 2015;43(Database issue):204–12.
12. Chatr-Aryamontri A, Breitkreutz BJ, Oughtred R, Boucher L, Heinicke S, Chen D, Stark C, Breitkreutz A, Kolas N, O'Donnell L. The BioGRID interaction database: 2015 update. Nucleic Acids Res. 2011;43(Database issue):D470.
13. Xenarios I, Rice DW, Salwinski L, Baron MK, Marcotte EM, Eisenberg D. DIP: the database of interacting proteins: 2001 update. Nucleic Acids Res. 2000; 32(1):D449.
14. Orchard S, Ammari M, Aranda B, Breuza L, Briganti L, Broackes-Carter F, Campbell NH, Chavali G, Chen C, Del-Toro N. The MIntAct project--IntAct as a common curation platform for 11 molecular interaction databases. Nucleic Acids Res. 2014;42:358–63.
15. 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(Database issue):321–7.
16. Breuer K, Foroushani AK, Laird MR, Chen C, Sribnaia A, Lo R, Winsor GL, Hancock REW, Brinkman FSL, Lynn DJ. InnateDB: systems biology of innate immunity and beyond—recent updates and continuing curation. Nucleic Acids Res. 2013;41(Database issue):D1228.
17. 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(7):1655.
18. Bailey TL, Gribskov M. Methods and statistics for combining motif match scores. Journal of computational biology a journal of computational. Mol Cell Biol. 1998;5(2):211–21.
19. Delorenzi M, Speed T. An HMM model for coiled-coil domains and a comparison with PSSM-based predictions. Bioinformatics. 2002;18(4):617–25.
20. Liang Y, Liu S, Zhang S. Prediction of protein structural classes for low-similarity sequences based on consensus sequence and segmented PSSM. Comput Math Methods Med. 2015;2015(2):1–9.
21. Wang J, Wang C, Cao J, Liu X, Yao Y, Dai Q. Prediction of protein structural classes for low-similarity sequences using reduced PSSM and position-based secondary structural features. Gene. 2015;554(2):241–8.
22. Chen K, Kurgan L. Computational prediction of secondary and Supersecondary structures: Humana Press; 2013.
23. Tomii K, Kanehisa M. Analysis of amino acid indices and mutation matrices for sequence comparison and structure prediction of proteins. Protein Eng. 1996;9(1):27.
24. Lobo I. Basic local alignment search tool (BLAST). J Mol Biol. 2008;215(3): 403–10.
25. Chen Z, Sun SK. A Zernike moment phase-based descriptor for local image representation and matching. IEEE transactions on image processing a publication of the IEEE signal processing Society 2010, 19(1):205–219.
26. Chong CW, Raveendran P, Mukundan R. A comparative analysis of algorithms for fast computation of Zernike moments. Pattern Recogn. 2003;36(3):731–42.
27. Farzam M, Shirani S. A robust multimedia watermarking technique using Zernike transform. In: Multimedia Signal Processing, 2001 IEEE Fourth Workshop on: 2001; 2001. p. 529–34.
28. Hse H, Newton AR. Sketched symbol recognition using Zernike moments. 2004;1:367–70.
29. Hwang SK, Billinghurst M, Kim WY. Local descriptor by Zernike moments for real-time Keypoint matching. In: Image and Signal Processing, Congress on: 2008; 2008. p. 781–5.
30. Khotanzad A, Hong YH. Invariant image recognition by Zernike moments. IEEE Trans Pattern Analys Mach Intell. 1990;12(5):489–97.
31. Kim WY, Kim YS. Sig Proc Image Commun. 2000;16(1–2):95–102.
32. Li S, Lee MC, Pun CM. Complex Zernike moments features for shape-based image retrieval. IEEE Trans Syst Man Cybernetics Part A Syst Hum. 2009; 39(1):227–37.
33. Liao SX, Pawlak M. On the accuracy of Zernike moments for image analysis. IEEE Trans Pattern Analys Mach Intell. 1998;20(12):1358–64.
34. Liao SX, Pawlak M. A study of Zernike moment computing; 2006.
35. Mukundan R, Ramakrishnan KR. Fast computation of Legendre and Zernike moments. Pattern Recogn. 1995;28(9):1433–42.
36. Noll RJ. Zernike polynomials and atmospheric turbulence. J Opt Soc Am. 1976;66(3):207–11 1917–1983.
37. Schwiegerling J, Greivenkamp JE, Miller JM. Representation of videokeratoscopic height data with Zernike polynomials. J Opt Soc Am A Opt Image Sci Vis. 1995;12(10):2105–13.
38. Singh C, Walia E, Upneja R. Accurate calculation of Zernike moments. Inf Sci. 2013;233(233):255–75.
39. Turney JL, Mudge TN, Volz RA. Invariant image recognition by Zernike moments. IEEE Trans Pattern Analys Mach Intell. 1990;12(5):489–97.
40. Wang JY, Silva DE. Wave-front interpretation with Zernike polynomials. Appl Opt. 1980;19(9):1510–8.
41. Mika S, Lkopf B, Smola A, Ller KR, Scholz M, Tsch G, Kernel PCA. de-noising in feature spaces. In: Conference on advances in neural information processing systems II: 1999; 1999. p. 536–42.
42. Sak H, Senior A, Beaufays F. Long short-term memory based recurrent neural network architectures for large vocabulary speech recognition. Com Sci. 2014:338–42.
43. Tai KS, Socher R, Manning CD. Improved semantic representations from tree-structured long short-term memory networks. Com Sci. 2015;5(1):36.
44. Dyer C, Ballesteros M, Ling W, Matthews A, Smith NA. Transition-based dependency parsing with stack long short-term memory. Com Sci. 2015; 37(2):321–32.
45. Wollmer M, Schuller B, Eyben F, Rigoll G. Combining long short-term memory and dynamic Bayesian networks for incremental emotion-sensitive artificial listening. IEEE J Selected Topics Signal Proc. 2010;4(5):867–81.
46. Sainath TN, Vinyals O, Senior A, Sak H. Convolutional, long short-term memory, fully connected deep neural networks. In: IEEE Int Conference on Acoustics, Speech and Signal Processing: 2015; 2015. p. 4580–4.
47. Srivastava N, Hinton G, Krizhevsky A, Sutskever I, Salakhutdinov R. Dropout: a simple way tprevent neural networks from overfitting. J Mach Learn Res. 2014;15(1):1929–58.
48. Dahl GE, Sainath TN, Hinton GE. Improving deep neural networks for LVCSR using rectified linear units and dropout. In: IEEE International Conference on Acoustics, Speech and Signal Processing: 2013; 2013. p. 8609–13.
49. Hinton GE, Srivastava N, Krizhevsky A, Sutskever I, Salakhutdinov RR. Improving neural networks by preventing co-adaptation of feature detectors. Com Sci. 2012;3(4):212–23.
50. Hanley JA, Mcneil BJ. The meaning and use of the area under a receiver operating characteristic (ROC) curve. Radiology. 1982;143(1):29.
51. Huang J, Ling CX. Using AUC and accuracy in evaluating learning algorithms. IEEE Trans Knowledge Data Eng. 2005;17(3):299–310.
52. Chang CC, Lin CJ. LIBSVM: a library for support vector machines. ACM Trans Intell Syst Technol. 2011;2(3):1–27.