Accepted Manuscript

Journal of Bioinformatics and Computational Biology

Article Title:	Protein secondary structure prediction improved by recurrent neural networks integrated with 2-dimensional convolutional neural networks
Author(s):	Yanbu Guo, Bingyi Wang, Weihua Li, Bei Yang
DOI:	10.1142/S021972001850021X
Received:	23 March 2018
Accepted:	26 July 2018
To be cited as:	Yanbu Guo *et al.*, Protein secondary structure prediction improved by recurrent neural networks integrated with 2-dimensional convolutional neural networks, *Journal of Bioinformatics and Computational Biology*, doi: 10.1142/S021972001850021X
Link to final version:	https://doi.org/10.1142/S021972001850021X

World Scientific
www.worldscientific.com

Journal of Bioinformatics and Computational Biology
© Imperial College Press

Protein secondary structure prediction improved by recurrent neural networks integrated with 2-dimensional convolutional neural networks

Yanbu Guo[1], Bingyi Wang[2], Weihua Li [1*], Bei Yang[3]

1 School of Information Science and Engineering, Yunnan University

No. 2 North Cuihu Road, Kunming 650091, P. R. China

2 The Research Institute of Resource Insects, Chinese Academy of Forestry

Bailongsi, Kunming 650224, P. R. China

3 MD. Cardiology Department, The Second People's Hospital of Yunnan Province

No.176 Qingnian Road, Kunming 650021, P. R. China

1 Email: guoyb213321@163.com

** Corresponding author' Email: liweihua@ynu.edu.cn*

2 Email: wbykm@aliyun.com

3 Email: kmyangbei@126.com

Abstract: Protein secondary structure prediction (PSSP) is an important research field in bioinformatics. The representation of protein sequence features could be treated as a matrix, which includes the amino-acid residue (time-step) dimension and the feature vector dimension. Common approaches to predict secondary structures only focus on the amino-acid residue dimension. However, the feature vector dimension may also contain useful information for PSSP. To integrate the information on both dimensions of the matrix, we propose a hybrid deep learning framework, 2-dimensional convolutional bidirectional recurrent neural networks (2C-BRNNs), for improving the accuracy of 8-class secondary structure prediction. The proposed hybrid framework is to extract the discriminative local interactions between amino-acid residues by 2-dimensional convolutional neural networks (2DCNN), and then further captures long-range interactions between amino-acid residues by bidirectional gated recurrent units (BGRUs) or bidirectional long short-term memory (BLSTM). Specifically, our proposed 2C-BRNNs framework consists of four models: 2DConv-BGRUs, 2DCNN-BGRUs, 2DConv-BLSTM and 2DCNN-BLSTM. Among these four models, the 2DConv- models only contain 2-dimensional (2D) convolution operations. Moreover, the 2DCNN- models contain 2D convolutional and pooling operations. Experiments are conducted on four public datasets. The experimental results show that our proposed 2DConv-BLSTM model performs significantly better than the benchmark models. Furthermore, the experiments also demonstrate that the proposed models can extract more meaningful features from the matrix of proteins, and the feature vector dimension is also useful for PSSP. The codes and datasets of our proposed methods are available at https://github.com/guoyanb/JBCB2018/.

Keywords: bioinformatics; protein secondary structure predication (PSSP); convolutional neural networks (CNNs); recurrent neural networks (RNNs); long short-term memory (LSTM); gated recurrent units (GRUs).

1 Introduction

1.1 Motivations

With the rapid development of proteomics, protein sequencing technologies have been resulted in enormous accumulation of protein sequences. Experimental methods, such as X-ray crystallography and nuclear magnetic resonance spectroscopy, are widely used to detect protein structures, because of their high precision. However, experimental methods are still time-consuming, expensive and labor-intensive, and may not always determine protein structures[1, 2]. In particular, experimental methods cannot meet the challenge of the rapid increase in protein sequences. Moreover, computational approaches of predicting protein structures have become more important and effective than before, and can handle large amounts of protein data and reduce the cost and time burden[1, 3].

Protein is one of the most important structural and functional macromolecules in living cells, which is involved in all the processes of life, such as the catalysis of biochemical reactions, the signal transduction [4]. The function of the protein is closely related to its structures[5, 6] , and is mainly determined by the protein's structures[6]. Structural understanding is critical not only for protein analysis, but also for the drug design[5, 7]. However, it's a great challenge for computational approaches to predicate protein structures by amino-acid sequences alone[3, 8]. Thus, accurate protein structure prediction is one of the most challenging tasks in computational biology [9-11].

Protein secondary structures refer to the local structure types of amino-acid sequences. According to the definition of secondary structure of protein (DSSP[12]) algorithm, protein secondary structures are divided into 3 classes or 8 classes [12, 13]. The amino-acid sequence and its corresponding 3 classes of secondary structures of PDB 154L are downloaded from http://www.rcsb.org/pdb/explore/explore.do?structureId=154L; and for every amino acid, the secondary structures of its neighbors are the most effective information to classify the secondary structure of this amino acid, as illustrated in Fig. 1 5.

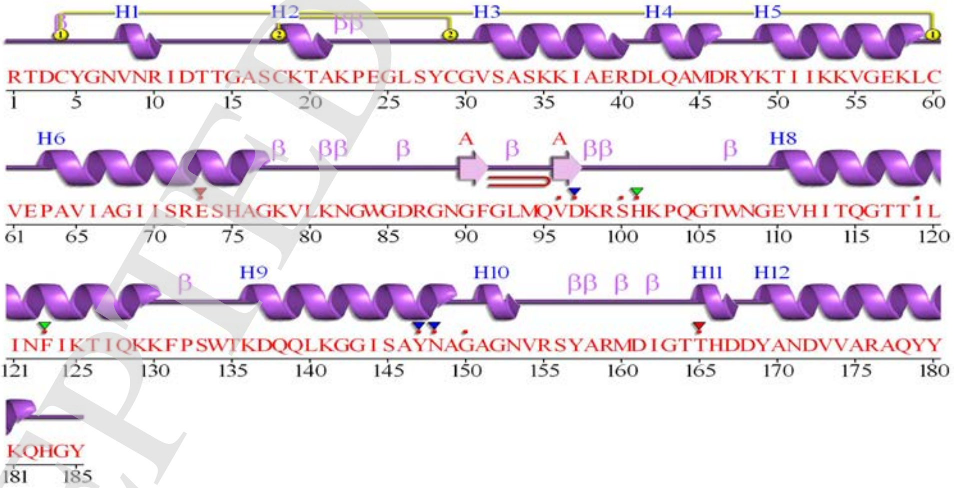


Fig. 1. The amino-acid sequence which consists of 185 amino-acid residues and its corresponding 3 classes of secondary structures of PDB 154L with UniProtKB accession number (P00718).

PSSP has been studied extensively for a long time[9, 13, 14] and many computational methods[1, 8] have been proposed to predict protein secondary structures. However, only a few methods were used to predict 8-class secondary structures, although 8-class secondary structures may provide more detailed local information. Moreover, the common approaches to predict secondary structures always utilize the non-sequential model, typically SVM and feed-forward neural networks[15-17]. These models are not ideal to identify secondary structures because protein sequences cannot be represented as a vector of fixed dimensionality[17]. In recent years, Asgari et al. [18] proposed a continuous distributed representation of biological sequences, called ProtVec[18], which is an informative and dense representation method based on deep learning models; and ProtVec could capture a diverse range of meaningful physical and chemical properties of biological sequences.

It's well known that recurrent neural networks (RNNs) and convolutional neural networks (CNNs) have shown good performance in various tasks, including text classification[19], image classification[20]. In natural language processing (NLP) tasks, CNNs usually utilize the 1D convolution operation and the 1D max pooling operation over the time-step dimension to obtain a fixed-length vector. However, in image processing tasks, CNNs apply 2D convolution and pooling operations to get a representation of the images over both the time-step dimension and the feature vector dimension.

Since the amino acid could be represented as a dense vector[4,18] in protein sequence, and then the protein sequences could be represent as a matrix, which includes the amino-acid residue (time-step) dimension and the feature vector dimension. However, in PSSP, the methods[5, 6, 11, 21] related with CNNs only utilize convolution operations and pooling operations along the amino-acid residue dimension of the matrices. And, the traditional convolution and pooling operations ignore the feature vector dimension[5, 11, 21] of the protein matrices, which could contain some useful information for PSSP ; meanwhile this may limit the performance of PSSP.

So, it's desirable to extract more discriminative features from the amino-acid residue dimension and the feature vector dimension of the protein matrices using 2D CNNs.

1.2 Related works

There are multiple levels in protein structure prediction, ranging from 1-dimension (1D) to 4-dimension (4D) structure prediction. The 1D structure prediction focuses on predicting the secondary structure and the solvent accessibility of each residue along the primary 1D protein sequence [1, 3]. In this study, the paper focuses on 8-class PSSP of the 1D structure prediction. The performance of PSSP methods is often assessed by Q3 accuracy or Q8 accuracy, which calculates the percent of residues for which secondary structures are correctly predicted.

In the 1970s, [1, 22] statistical models were frequently utilized to analyze the probability of amino acids appearing in protein secondary structures. The Q3 accuracy of these models was lower than 60% due to inadequate features. Later on, the Q3 accuracy exceeded 70% by taking advantage of the evolutionary information of protein[2] and position-specific scoring matrices (PSSM) [15]. Recently, Q3 accuracy [5, 21] has gradually been improved to above 80%. However, when these traditional models were used to predicate 8-class secondary structures, Q8 accuracy was very low, because they needed to distinguish among 8-class secondary structures [5, 6, 17, 21]. Thus, 8-class secondary structure prediction is more promising and attractive[1, 6].

In recent years, deep neural networks (DNNs) have become an established framework for the representation learning of various data[7, 8, 18, 23], and DNNs have also been proved to be effective and significantly improve accuracy on the 8-class secondary structure prediction[21]. Wang et al. [11] applied deep CNNs with a conditional random field to secondary structure prediction, and achieved 68.3% Q8 accuracy and 82.3% Q3 accuracy on the benchmark CB513 dataset. Li et al. [5] used a multi-scale convolutional layer followed by three stacked bidirectional recurrent layers, and achieved 69.7% Q8 accuracy on the same test dataset. Busia et al. [24] used a novel chained CNNs and next-step conditioning to achieve 71.4% Q8 accuracy on the same test dataset. Lin et al. [25] proposed a deep convolutional neural network architecture combined with a multilayer shift-and-stitch for PSSP, and achieved 68.4% Q8 accuracy on the same test dataset. Pollastri et al. [26] utilized RNNs and profiles to improve 8-class PSSP, and achieved 51.1%[17] Q8 accuracy on the same test dataset. Sønderby et al. [17] used a bidirectional recurrent neural networks (BRNNs) with long short-term memory cells for PSSP and achieved 67.4% Q8 accuracy on the same test dataset. Thus, PSSP[3, 15, 27] based on deep learning methods has been followed with great attention by researchers in proteomics.

Inspired by the recent successes of DNNs, we propose a hybrid deep learning framework, 2-dimensional convolutional bidirectional recurrent neural networks (2C-BRNNs) to improve the accuracy of 8-class PSSP.

This framework integrates 2D convolutional or 2D pooling operations with bidirectional GRUs and LSTM, including four models: 2DConv-BGRUs, 2DConv-BLSTM, 2DCNN-BGRUs, and 2DCNN-BLSTM. The former two models only contain 2D convolution operations, while the latter two models consist of 2D convolution operations and 2D pooling operations.

1.3 Ideas and contribution*s*

PSSP is a bridge between the primary and the tertiary structure, and protein secondary structures are the early folding stage of protein molecule as the foundation of protein 3-D structure[28]. Structural understanding of proteins is the fundament of protein analysis[7], because the secondary structures can help to understand the relationship between the function and primary structure [10, 28]. Local features of the amino-acid sequences are the critical and effective features for assigning the secondary structure of an amino acid, because the adjacent amino acids may belong to the same class of secondary structures. In PSSP, these local interactions can be extracted by CNNs [11, 21], which use 1D convolutional operations. In addition, CNNs with 2D convolutional and pooling operations also have the potential to capture the temporal and spatial features in image classification tasks[20, 29]. Inspired by that, we represent protein features as a matrix, and utilize 2-dimensional CNNs to extract the local interactions between amino-acid residues.

In addition, the long-range interactions [5, 14] between amino acids also hold discriminative features for PSSP. RNNs are useful to model sequence data and can capture long-range interactions of sequence data. However, RNNs couldn't perform well on PSSP in previous work[26], partially due to the difficulty to train such models. Fortunately, RNNs with gate structures, especially GRUs[30] and LSTM[31], can be more effective for protein sequences because these models can artificially remember and forget information by gate structures. Thus, this paper applies GRUs and LSTM to extract long-range interactions between amino-acid residues in protein sequences.

In summary, the main contributions of this paper are as follows: (1) This paper proposes a novel deep learning framework 2C-BRNNs, which utilizes 2D CNNs to extract local interactions between amino-acid residues, and stacked BGRUs or BLSTM to extract long-range interactions between amino-acid residues. As far as we know, this work first combines 2D CNNs with RNNs with GRUs and LSTM to 8-class PSSP. (2) Our proposed framework is verified on the CB6133, filtered CB6133, CB513, CASP10 and CASP11 datasets. Experimental results show that our framework outperforms existing methods and achieves state-of-the-art performance. In addition, it also suggests the feature vector dimension of protein matrices contributes to predicting secondary structures (3) To better understand the effect of different RNNs units and different 2D convolutional filters on our proposed models, we conduct the additional experiment on the public datasets mentioned above. It depicts the performance of the proposed models on different RNNs units and convolutional filters.

2 Proposed Methods

As shown in Fig. 2, the proposed framework comprises four parts: the dense feature layer, the local interactions extractor (2D convolutional layer), the long-range interactions extractor (bidirectional GRUs or LSTM recurrent neural networks) and the output layer for 8-class PSSP. Moreover, the output layer comprises the feature fusion layer and the fully connected layer.

To extract the global features of proteins, the local interactions of amino acids and the long-range interactions of amino acids are fed into the feature fusion layer. Finally, the output from the feature fusion layer is fed into the predictor with the softmax activation, which achieves 8-class secondary structure prediction.

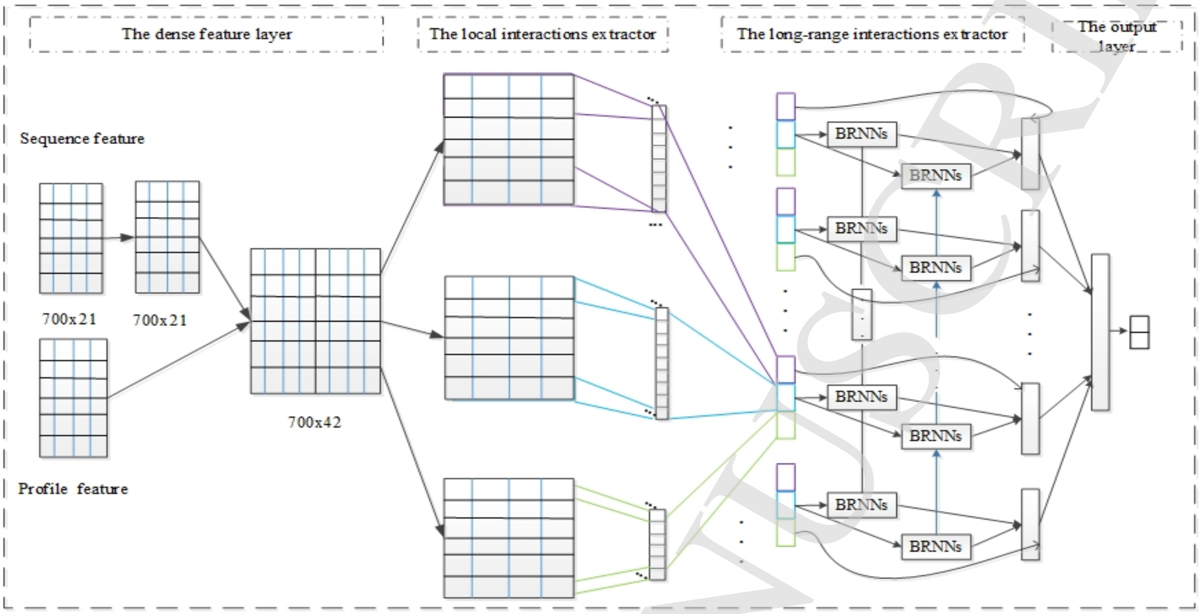


Fig. 2. Our proposed hybrid framework for predicting 8-class protein secondary structures

2.1 The dense feature layer

Protein is a type of macromolecule made of amino-acid building blocks arranged in a linear chain and concatenated simultaneously by peptide bonds[1] . The linear polypeptide chain is named after the primary structure of protein. As shown in Fig. 2, the initial representation of protein features consists of the sequence feature and the profile feature. The dense feature layer is used to transform the sparse sequence feature vector into the dense feature vector.

In this paper, we represent the features of the protein sequence as a dense 42-dimensional vector, which is decomposed into two parts. One is 21-dimensional sequence features, which encode the types of the amino acids; and the other is 21-dimensional profile features, which are obtained from the PSI-BLAST log file[4] and rescaled by a logistic function[15] . Every sequence feature vector is an one-hot vector, which all bits are “0” except one “1” in the vector. In addition, every profile feature vector has a dense representation. Simultaneously, as Li et al. [5] did, to avoid the inconsistency in feature representations of proteins, we also transform sparse sequence features into the dense vector by an embedding operation[23]. Actually, the embedding operation can map the 21-dimensional sparse vector into the 21-dimensional dense vector. Finally, both the dense sequence feature and the profile feature are concatenated as the initial protein feature $p=\{p_1, p_2, p_3 \cdots, p_n\}$, $p_i \in \mathbb{R}^{42}$. The secondary structure label of an amino-acid can be represented as $L=\{L_1, L_2, \cdots, L_n\}$, $L_i \in \mathbb{R}^8$.

2.2 The local interactions extractor

To make full use of the local interactions of amino-acid residues in PSSP, we apply 2D convolution operations and 2D pooling operations to capture local interactions over two dimensions of protein matrices.

Supposed that the matrix $p=\{p_1, p_2, p_3 \cdots, p_n\}$ is obtained from the dense feature layer, where $p \in \mathbb{R}^{n\times k}$, and p_i is the preprocessed feature vector of the i_{th} amino-acid residue in sequences. In this layer, the 2D convolution filter $F \in \mathbb{R}^{f_1 \times f_2}$ is first applied to the window of f_1 amino-acid residues and f_2 feature vectors with the activation function ReLu (rectified linear unit), as shown in Equation 1. B is the bias term and $\odot$ represents dot product.

$$c_{i,j} = Re\,Lu\left(F \odot p_{i:i+f_1-1,:j+f_2-1} + B\right) \tag{1}$$

This filter F is applied to each possible window of the matrix p to produce a feature map c, as shown in Equation 2.

$$c = \left[c_{1,1}, c_{1,2}, \cdots, c_{n-f_1+1,k-f_2+1}\right] \tag{2}$$

In Equation 2, it describes the process of the filter. The convolutional layer can have multiple filters with different size filters to learn more complementary interactions.

Then, the 2D max-pooling $m \in \mathbb{R}^{q_1 \times q_2}$ operation is further applied to extract the maximum value over the window of the matrix c, as shown in Equation 3.

$$m_{i,j} = f\left(c_{i:i+q_1,\,j:j+q_2}\right) \tag{3}$$

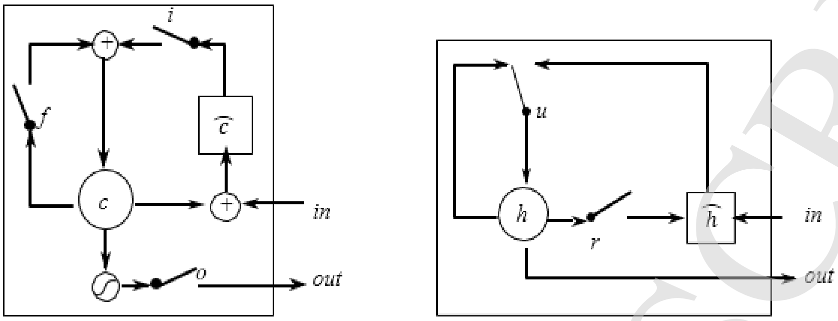
In Equation 3, $f(\cdot)$ represents the 2D max-pooling function. Then the pooling results are shown as Equation 4:

$$m = \left[m_{1,1}, m_{1,1+q_2}, \cdots, m_{1+(n-f_1+{}^{1}\!/_{q_1}-1)\bullet q_1,1+(k-f_2+{}^{1}\!/_{q_2}-1)\bullet q_2}\right] \tag{4}$$

2.3 The long-range interactions extractor

On the other hand, there are also the long-range interactions except for local interactions in protein sequences, and these long-range interactions also affect the class of the secondary structures of amino-acid residues. Because 2D convolution operations and the 2D pooling operations have limited kernel sizes, some interactions between amino-acid residues may not be extracted by them. Thus, we further utilize bidirectional recurrent neural networks with GRUs or LSTM to capture the long-rang interactions.

Although RNNs[23, 26] have the great ability to deal with sequence data, it is difficult to train RNNs due to the vanishing gradient problem. The RNNs with LSTM and GRUs can avoid this problem while learning the long-range interactions in sequence data. Then, the structure unit is shown in Fig. 3. Thus, this paper exploits BRNNs with GRUs and LSTM to capture the long-range interactions between amino-acid residues.



(a) Long Short-Term Memory units (b) Gated Recurrent Units

Fig. 3. The structure of the LSTM unit and GRUs

The LSTM unit has the memory cell *c*, one output state and three gates, namely input gate *i*, forget gate *f* and output gate *o* respectively, as shown in Fig. 3(a). Compared with the LSTM, the GRUs merge the input gate and the forget gate together to form the update gate *u* and the reset gate *r*, and they can control what is artificially remembered or forgotten.

Provided the input feature is represented as (m_t , h_{t-1}) at time *t*, the update of the LSTM unit can be represented as following Equation (5-9).

$$f_t = sigmoid\left(W_{xf} m_t + W_{hf} m_{t-1} + B_f\right) \tag{5}$$

$$i_t = sigmoid\left(W_{xi} m_t + W_{hi} m_{t-1} + B_i\right) \tag{6}$$

$$c_t = f_t \odot c_{t-1} + i_t \odot tanh\left(W_{xc} m_t + W_{hc} m_{t-1} + B_c\right) \tag{7}$$

$$o_t = sigmoid\left(W_{xo} m_t + W_{ho} m_{t-1} + B_o\right) \tag{8}$$

$$h_t = o_t \odot tanh\left(c_t\right) \tag{9}$$

In Equation (5-9), f_t , i_t and o_t is the activation of the forget gate, input gate and output gate respectively. Moreover, c_t is the current cell state; W is the weight matrix and B is the bias term. In addition, $\odot$, $sigmoid(\ )$ and $tanh(\ )$ represents the element-wise multiplication, the sigmoid and hyperbolic functions respectively.

The mechanism of GRUs is shown in Fig. 3(b). Provided the input is represented as (m_t , h_{t-1}) at time *t*, the update of the GRUs is formalized as Equation (10-12).

$$r_t = sigmoid\left(W_{xr} m_t + W_{hr} m_{t-1} + B_r\right) \tag{10}$$

$$u_t = sigmoid\left(W_{lu} m_t + W_{hu} m_{t-1} + B_u\right) \tag{11}$$

$$\widehat{h_t} = tanh\left(W_{l\widehat{h}} m_t + W_{h\widehat{h}}\left(r_t \odot h_{t-1} + B_{\widehat{h}}\right)\right) \tag{12}$$

$$h_t = u_t \odot h_{t-1} + \left(1 - u_t\right) \odot \widehat{h_t} \tag{13}$$

In Equation (10-13), r_t , u_t , h_t and $\widehat{h_t}$ is the activation of the reset gate, update gate, GRUs output, internal memory cell respectively.

Moreover, the secondary structure of an amino acid at any position not only depends on the preceding amino acids, but also the succeeding amino acids in protein sequences. Thus, this paper utilizes bidirectional RNNs (BRNNs) to predict the secondary structure and designs two types of models. The first model consists of a forward GRUs and a backward GRUs, and the second model consists of a forward LSTM and a backward LSTM.

We take the BRNNs model as an example to illustrate how these models capture the long-range interactions between amino-acid residues. At time *t*, the output feature from 2D convolutional layers is fed into the BRNNs to extract the forward and backward information of amino-acid residues. The forward and backward hidden states of the BRNNs layer are calculated according to Equation (14-15), where *z* and $h_t^{z\text{-}1}$ denotes the layer index and the concatenated output feature of the preceding layer in the stacked BRNNs respectively. At last, the obtained feature is the concatenation of these two parts according to Equation 16; *R* represents *c* or *m*; *c* is the features extracted by 2D convolutional operations in Equation (1-2); *m* is the features extracted by 2D convolutional operations and max-pooling operations in Equation (1-3).

$$\overrightarrow{h_t^z} = BRNNs\left(h_t^{z-1}, \overrightarrow{h_{t-1}^z}\right) \tag{14}$$

$$\overleftarrow{h_t^z} = BRNNs\left(h_t^{z-1}, \overleftarrow{h_{t+1}^z}\right) \tag{15}$$

$$h_t = \left[\overrightarrow{h_t^z}; \overleftarrow{h_t^z}; \mathrm{R}\right] \tag{16}$$

Besides, two BRNNs with different units (GRUs or LSTM) are stacked together to improve the performance. When the BGRUs model is utilized to capture the long-range interactions between amino-acid residues, the hidden units are updated by Equation (10-13), and *BRNNs*( ) of Equation (14-15) represents the BGRUs model. In addition, when the BLSTM model is utilized to capture the long-range interactions between amino-acid residues, the hidden units are updated by Equation (5-9), and *BRNNs*( ) of Equation (14-15) represents the BLSTM model.

2.4 The output layer

Our proposed models extract the protein feature, and the feature is record as $h = \left[h_1, h_2, \cdots, h_T\right]$. The protein feature is first fed into a fully connected layer, and the softmax layer is used to predict the probability of the class of an amino-acid residue, as shown in Equation (17-18).

$$p_i\left(y \mid s\right) = soft\,max\left(w^s h + b^s\right) \tag{17}$$

$$soft\,max\left(z\right) = \frac{e^z}{\sum e^z} \tag{18}$$

In experiments, we train our proposed models by the stochastic gradient descent algorithm Adm[32] and the error backward propagation. The training goal is to minimize the cross-entropy loss function (see Equation 19). Meanwhile, all the parameters are adjusted according to Equation 20, and θ is the parameter set; α is the learning rate, and λ is the L2 regularization hyper-parameter.

$$L\left(\theta\right) = -\frac{1}{N}\sum_{k=1}^{N}\sum_{j=1}^{c} y_{kj}\left(log\left(p_{kj}\right)\right) + \lambda\left\|\theta\right\|^2 \tag{19}$$

$$\theta \leftarrow \theta + \alpha \frac{\partial L\left(\theta\right)}{\partial \theta} \tag{20}$$

3 Experimental datasets and evaluation metrics

3.1 Datasets

To evaluate our proposed models, we compare the results with other published results on four public datasets: CB6133, CB513, CASP10 and CASP11.

(1) CB6133[6] is produced with PISCES CullPDB[33] system. This dataset has 6128 proteins, of which 5600 are for training, 256 for validation and 272 for testing.

(2) CB513[6] contains 513 proteins and is widely used as a secondary structure dataset for the performance comparison.

The CB6133 and CB513 datasets are preprocessed by Zhou et al.[6], and these datasets can be available at http://www.princeton.edu/~jzthree/datasets/ICML2014/.

(3) CASP10[34] and CASP11[35] contains 123 and 105 domain sequences respectively. Critical assessment of protein structure prediction (CASP) is a community-wide protein structure prediction competition. These datasets are also widely used for PSSP.

CB6133 is a non-homologous protein structure and sequence dataset. This paper uses the CB6133 dataset to train and test the deep neural models. The CB513 dataset is a public benchmark dataset only used for testing. However, there exists redundancy between the CB513 dataset and the CB6133 dataset. A filtered version of the CB6133 dataset is generated by removing sequences that have over 25% similarity between CB6133 and CB513. The filtered CB6133 dataset contains 5534 proteins. When PSSP is evaluated on the (2-3) datasets, 5278 protein sequences were randomly chosen to train from the filtered CB6133, and the remaining 256 protein sequences were to test.

Every protein sequence in the aforementioned datasets has 57 channels. The '[0,22)' channel is the amino-acid residue; The '[22,31)' channel is the secondary structure label; The '[31,33)' channel is N- and C-terminals; The '[33,35)' channel is the relative and absolute solvent accessibility; The '[35,37)' channel is the sequence profile, which is generated by rescaling PSSM through a logistic function (PSSM is derived by PSI-BLAST against the UniRef90 database with E-value threshold 0.001 and 3 iterations)[6]. 8 channels are the secondary structure labels. 2 channels are the solvent accessibility labels.

The sequence feature and the sequence profile are both encoded as $n \times b$ matrices, where *n* is protein length and *b* is the number of amino-acid types. In order to compare these existing methods conveniently, the channel length is also set to 700 in our proposed models. This is similar to the existing researches[5, 6]. Because of the majority of protein sequences are less than 700 amino acids, the 700 amino-acid length cutoff is chosen to provide a good balance between efficiency and coverage. In other words, the length of sequences longer than 700 are truncated and the length of sequences shorter than 700 are padded with zeros.

3.2 Evaluation methods

PSSP is most commonly evaluated by Q3 accuracy and Q8 accuracy, which give the percentage of residues correctly predicted for 3 and 8 protein secondary structure classes respectively. Due to focusing on the 8-class PSSP, this paper adopts Q8 accuracy as the primary performance measures. Besides, this paper also adopts the Loss value as the secondary performance measures. The Loss value is the cross entropy value between the predicted value and the truth label, as shown in Equation (19).

4 Experimental Results

In this section, we analyze the experimental results of our proposed models for 8-class PSSP on four publicly available datasets: CB6133, CB513, CASP10 and CASP11. Results show that the proposed models can achieve the excellent performance.

4.1 Experimental settings and Implementation

Our codes are implemented in Keras (https://keras.io/) library, which is a high-level neural networks API based on Tensorflow (https://www.tensorflow.org/). Weights of our proposed models are initialized with the default setting in Keras. We simultaneously train all the layers with the Adam optimizer. The batch size is 64. The proposed models are trained on a single NVIDIA GeForce GTX 1060 GPU with 6GB memory. Meanwhile, we use the early stopping and the dropout method to avoid over-fitting.

In experiments, the width and height of the 2D convolution filter are 3. Each feature map has 42 channels. To avoid over-fitting, the local interactions of proteins are fed into the fully connected layer with 400 hidden units and then are regularized with dropout (0.4); the features of the fully connected layer are passed into the two stacked BRNNs layers, each with 500 hidden units. Moreover, the local and long-range interactions are regularized with dropout (0.5). Finally, the local and long-range interactions are fed into the fully connected layer of 600 hidden units with the ReLu activation.

4.2 Overall Performance

We perform two sets of experiments on four public datasets. In the first experiment, we perform both training and testing on the original CB6133 dataset. In the second experiment, we perform training on the filtered CB6133 dataset and testing on the CB513, CASP10 and CASP11 dataset.

Table 1 presents the performance of our proposed models on four public datasets. The 2DConv-BLSTM model achieves the most excellent performance on datasets. Especially, this model achieves 75.7% Q8 accuracy and 74.5% Q8 accuracy on CB6133 and CASP10 respectively.

Table 1. The overall performance of our proposed models for 8-class secondary structure prediction on four public datasets

Our Methods	**CB6133**		**CB513**		**CASP10**		**CASP11**	
	Q8 (%)	**LOSS (%)**	**Q8 (%)**	**LOSS (%)**	**Q8 (%)**	**LOSS (%)**	**Q8 (%)**	**LOSS (%)**
2DConv-BGRUs	74.9	21.5	69.0	20.8	72.2	19.7	71.0	21.7
2DCNN-BGRUs	73.5	23.1	68.7	20.9	72.1	19.8	71.7	21.2
2DConv-BLSTM	**75.7**	21.5	**70.2**	20.2	**74.5**	18.1	72.5	20.4
2DCNN-BLSTM	74.3	21.1	70.0	20.5	74.5	18.2	72.6	20.6

In addition, we also compare our models against other existing methods (DeepCNF-SS8[30], CNF-SS8[31], SSpro8-SS8[23]and GSN-SS8[6]) on four public datasets. In this comparison, all models except CNF-SS8 were trained using the standard sequence feature and profile feature. Besides the standard sequence feature and profile feature, three extra features were used in CNF-SS8 model, and the feature vector of an amino acid is 78-dimensional. Even so, our models still achieve a higher Q8 accuracy, as shown in Table 2 and Table 3.

(1) Training and Testing on CB6133

We compare our models with the four benchmark models on CB6133 by Q8 accuracy. All these models are trained and tested on the CB6133 dataset.

According to Table 2, the performance of our models exceeds the benchmark models. Specifically, our **2DConv-BLSTM** model achieves the highest Q8 accuracy of 75.7%, an increase of 0.5% compared to the DeepCNF-SS8 model. In addition, the Q8 accuracy of our other three models is lower than that of the DeepCNF-SS8 model, but higher than that of the SSpro-SS8 (without template), RaptorX-SS8 and GSN model**.** We think that the better performance owes to the power of our hybrid model integrated **2D convolution**, **2D pooling** with stacked bidirectional **LSTM**, and also owes to the local and long-range interactions captured by this model. The result suggests that our proposed **2DConv-BLSTM model** is more expressive in protein sequences.

Table 2. The overall Q8 accuracy of our models and the benchmark models on the CB6133 dataset.

Models	**Methods**	**Q8(%)**
Benchmark models	SSpro-SS8 (without template)	66.6
	RaptorX-SS8	69.7
	GSN-SS8	72.1
	DeepCNF-SS8	75.2
Our models	**2DConv-BGRUs**	74.9
	2DConv-BLSTM	**75.7**
	2DCNN-BGRUs	73.5
	2DCNN-BLSTM	74.3

(2) Training on Filtered CB6133 and Testing on CB513, CASP10 and CASP11

We compare our models with the four benchmark models by Q8 accuracy. All the models are trained on the filtered CB6133 and tested on the public CB513, CASP10 and CASP11 datasets.

According to Table 3, in our proposed models, 2DConv-BLSTM and 2DCNN-BLSTM perform better than the state-of-the-art models and achieve Q8 accuracy more than 70% on four public datasets. Especially, 2DConv-BLSTM achieves 70.2% Q8 accuracy and higher 0.9 than the DeepCNF-SS8 on the CB6133. 2DConv-BLSTM and 2DCNN-BLSTM achieve 74.5% accuracy and higher 2.7 than DeepCNF-SS8 on the CASP10. 2DCNN-BLSTM achieves 72.6% accuracy and higher 0.3 than DeepCNF-SS8 on the CASP11. In addition, 2DCNN-BGRUs and 2DConv-BGRUs achieve the Q8 accuracy less than 70% on the CB6133 dataset, but higher than the benchmark models on the CASP10 and the CB513 dataset. The Q8 accuracy of the 2DConv-BGRUs is lower 1.3 than that of the DeepCNF-SS8 on the CASP11 dataset. The Q8 accuracy of the 2DCNN-BGRUs is lower 0.5 than that of the DeepCNF-SS8 on the CASP11 dataset.

In brief, our models achieve excellent performance on CB513, the CASP10 and CASP11 dataset. This fully demonstrates our deep learning framework is efficient and practical on the 8-class PSSP.

Table 3. The overall Q8 accuracy of our models compared with the benchmark models on the public CB513, CASP10 and CASP11 dataset.

Models	**Methods**	**CB513(%)**	**CASP10(%)**	**CASP11(%)**
Benchmark models	SSpro-SS8 (without template)	63.5	64.9	65.6
	GSN-SS8	66.4	—	—
	RaptorX-SS8	64.9	64.8	65.1
	DeepCNF-SS8	68.3	71.8	72.3
Our models	**2DConv-BGRUs**	69.0	72.2	71.0
	2DConv-BLSTM	**70.2**	**74.5**	72.5
	2DCNN-BGRUs	68.7	72.1	71.7
	2DCNN-BLSTM	70.0	**74.5**	**72.6**

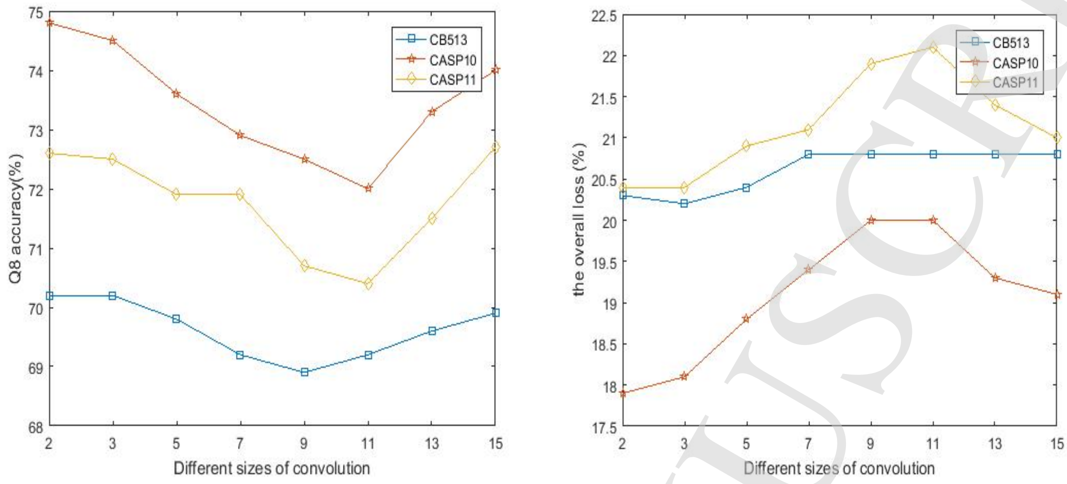
4.3 Effect of 2D convolution filter

Taking the 2DConv-BLSTM model as an example on the three public datasets, this section reveals the 2D convolutional filter has some effect on the performance of the 2DConv-BLSTM model. The details of the experiments are shown in Table 4. We conduct experiments with different convolution sizes from 2 to 15 on the CB513, CASP10 and CASP11 datasets. Table 4 shows the performance of 2DConv-BLSTM with different 2D convolution filters.

Table 4. The overall Q8 performance of the 2DConv-BLSTM model with different convolution sizes on the CB513, CASP10 and CASP11 dataset.

Sizes of convolution operations	**CB513**		**CASP10**		**CASP11**	
	Q8 (%)	**LOSS (%)**	**Q8 (%)**	**LOSS (%)**	**Q8 (%)**	**LOSS (%)**
2	**70.2**	20.3	**74.8**	17.9	**72.6**	20.4
3	**70.2**	20.2	**74.5**	18.1	72.5	20.4
5	69.8	20.4	73.6	18.8	71.9	20.9
7	69.2	20.8	72.9	19.4	71.9	21.1
9	68.9	20.8	72.5	20.0	70.7	21.9
11	69.2	20.8	72.0	20.0	70.4	22.1
13	69.6	20.8	73.3	19.3	71.5	21.4
15	69.9	20.8	74.0	19.1	72.7	21.0

Fig. 4 depicts that the model with different convolution sizes achieves different Q8 accuracy and loss value on the three public datasets. In Fig. 4, the x-axis represents the convolution length of the filter and y-axis is the Q8 accuracy or the overall loss. From Fig. 4(a), under the LSTM unit size 250, the best Q8 accuracy on three public dataset is respectively 70.2%, 74.8% and 72.6% with 2D filter size (2, 2). From Fig. 4(b), the loss value increases first and then decreases when the hidden units increase gradually. This shows that the finer tuning can further improve the performance.



(a) The Q8 accuracy on three public datasets (b) The overall loss on three public datasets

Fig. 4. The performance of the 2DConv-BLSTM model with different convolution sizes on the CB513, CASP10 and CASP11 dataset

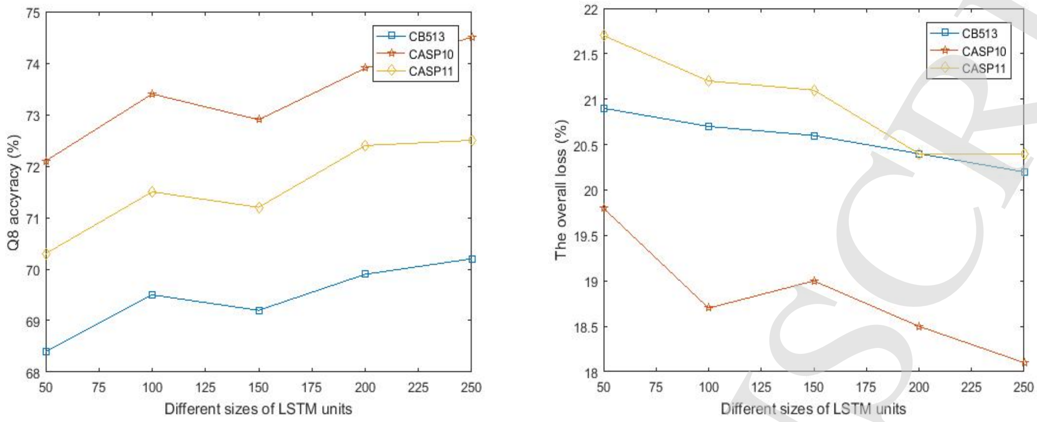
4.4 Effect of the BRNNs unit size

Taking the 2DConv-BLSTM model as an example, this section is to illustrate the effect of the BRNNs unit size on the model on CB513, CASP10 and CASP11 dataset. We conduct experiments with different sizes of the hidden units from 50 to 250 on three public datasets. The details of experiments are shown in Table 5. Table 5 shows the performance of 2DConv-BLSTM with different sizes of LSTM units.

Table 5. The overall Q8 performance of the **2DConv-BLSTM** model with different sizes of LSTM units on the CB513, CASP10 and CASP11 dataset.

Sizes of LSTM units	**CB513**		**CASP10**		**CASP11**	
	Q8 (%)	**LOSS (%)**	**Q8 (%)**	**LOSS (%)**	**Q8 (%)**	**LOSS (%)**
50	68.4	20.9	72.1	19.8	70.3	21.7
100	69.5	20.7	73.4	18.7	71.5	21.2
150	69.2	20.6	72.9	19.0	71.2	21.1
200	69.9	20.4	73.9	18.5	72.4	20.4
250	**70.2**	**20.2**	**74.5**	**18.1**	**72.5**	**20.4**

Fig. 5 depicts that the 2DConv-BLSTM model with different hidden units of the BLSTM can achieve different accuracies on three public datasets. In Fig. 5(a) and Fig. 5(b), the x-axis represents the convolution length of the filter and y-axis is Q8 accuracy or the overall loss value. Under the convolution size (3,3), the best Q8 accuracy is respectively 70.2%, 74.5% and 72.5% with the LSTM unit size 250 on the CB513, CASP10 and CASP11 dataset. From Fig. 5(b), the loss value decreases gradually when the hidden units increase. Those show that the larger LSTM hidden unit can detect the more long-range interactions.



(a) The Q8 accuracy on three public datasets (b) The overall loss on three public datasets

Fig. 5. The performance of the 2DConv-BLSTM model with different sizes of LSTM units on the CB513, CASP10 and CASP11 dataset

5 Conclusions

CNNs and RNNs with gated units (LSTM and GRUs) have become efficient and effective in various fields. To extract more discriminative features from protein matrices for 8-class PSSP, we propose the hybrid deep learning framework, bidirectional recurrent neural networks integrated with 2-dimensonal convolutional neural networks. By integrating the information on both dimensions of the matrix, and meanwhile integrating the local and long-range interactions between amino-acid residues, the performance of previous methods for PSSP has been improved. Additional experiments show the higher Q8 accuracy can be achieved by increasing hidden units. Moreover the experiments suggest that the feature vector dimension of protein matrices contributes to predicting protein secondary structures. Based on the success of our proposed deep neural networks on secondary structure prediction, we deduce that such models can also be applied to other challenging tasks in proteomics.

Existing BGRUs or BLSTM cannot deal with low frequency long-range interactions and the imbalance problem of PSSP. Outstanding neural architectures with the attention mechanism may be appropriate to solve this problem; In future work, we hope to apply the attention mechanism to the study of low frequency long-range interactions and the imbalance problem of PSSP.

Acknowledgments

We appreciate those constructive works: Zhou et al.[6] and Li et al.[5] provided the four public datasets. Besides, we would like to thank the anonymous reviewers for their valuable comments. This work was partially funded by Personnel Training Program of Academic and Technical Leaders of Yunnan Province (Bingyi Wang), National Science Foundation of China (Grant NO.11661081) , Integration of Cloud Computing and Big Data, Innovation of Science and Education (2017B00016).

References

1. Cheng J, Tegge AN, Baldi P. Machine learning methods for protein structure prediction. *IEEE reviews in biomedical engineering* **1:** 41-49, 2008.
2. Rost B, Sander C. Prediction of protein secondary structure at better than 70% accuracy. Journal of molecular biology 232(2): 584-599, 1993.

3. Min S, Lee B, Yoon S. Deep learning in bioinformatics. Briefings in bioinformatics 18(5): 851-869, 2017.

4. Altschul SF, Madden TL, Schäffer AA, Zhang J, Zhang Z, Miller W et al. Gapped BLAST and PSI-BLAST: a new generation of protein database search programs. Nucleic acids research 25(17): 3389-3402, 1997.

5. Li Z, Yu YZ. Protein secondary structure prediction using cascaded convolutional and recurrent neural networks. Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence. AAAI Press, 2560-2567, 2016.

6. Zhou J, Troyanskaya OG. Deep supervised and convolutional generative stochastic network for protein secondary structure prediction. arXiv preprint arXiv:1403.1347, 2014.

7. Gawehn E, Hiss JA, Schneider G. Deep learning in drug discovery. Molecular informatics 35(1): 3-14, 2016.

8. Angermueller C, Pärnamaa T, Parts L, Stegle O. Deep learning for computational biology. Molecular systems biology 12(7): 878, 2016.

9. Qian N, Sejnowski TJ. Predicting the secondary structure of globular proteins using neural network models. Journal of molecular biology 202(4): 865-884, 1988.

10. Wang S, Li W, Liu S, Xu J. RaptorX-Property: a web server for protein structure property prediction. Nucleic acids research 44(W1): W430-W435, 2016.

11. Wang S, Peng J, Ma J, Xu J. Protein secondary structure prediction using deep convolutional neural fields. Scientific reports 6: 18962, 2016.

12. Kabsch W, Sander C. Dictionary of protein secondary structure: pattern recognition of hydrogen-bonded and geometrical features. Biopolymers 22(12): 2577-2637, 1983.

13. Pauling L, Corey RB. Configurations of polypeptide chains with favored orientations around single bonds two new pleated sheets. Proceedings of the National Academy of Sciences 37(11): 729-740, 1951.

14. Heffernan R, Yang Y, Paliwal K, Zhou Y. Capturing non-local interactions by long short-term memory bidirectional recurrent neural networks for improving prediction of protein secondary structure, backbone angles, contact numbers and solvent accessibility. Bioinformatics 33(18): 2842-2849, 2017.

15. Jones DT. Protein secondary structure prediction based on position-specific scoring matrices1. Journal of molecular biology 292(2): 195-202, 1999.

16. Hua S, Sun Z. A novel method of protein secondary structure prediction with high segment overlap measure: support vector machine approach1. Journal of molecular biology 308(2): 397-407, 2001.

17. Sønderby SK, Winther O. Protein secondary structure prediction with long short term memory networks. arXiv preprint arXiv:1412.7828, 2014.

18. Asgari E, Mofrad MR. Continuous distributed representation of biological sequences for deep proteomics and genomics. PloS one 10(11): e0141287, 2015.

19. Zhou P, Qi ZY, Zheng SC, Xu JM, Bao HY, Xu B. Text Classification Improved by Integrating Bidirectional LSTM with Two-dimensional Max Pooling. Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics 3485-3495, 2016.

20. Wei Y, Xia W, Lin M, Huang J, Ni B, Dong J et al. Hcp: A flexible cnn framework for multi-label image classification. IEEE transactions on pattern analysis and machine intelligence 38(9): 1901-1907, 2016.

21. Fang C, Shang Y, Xu D. MUFOLD-SS: New deep inception-inside-inception networks for protein secondary structure prediction. Proteins: Structure, Function, and Bioinformatics 86(5):592-598, 2018.

22. Chen H, Gu F, Huang Z. Improved Chou-Fasman method for protein secondary structure prediction. BMC bioinformatics 7(4): S14, 2006.

23. Mesnil G, Dauphin Y, Yao K, Bengio Y, Deng L, Hakkani-Tur D et al. Using recurrent neural networks for slot filling in spoken language understanding. IEEE/ACM Transactions on Audio, Speech, and Language Processing 23(3): 530-539, 2015.

24. Busia A, Jaitly N. Next-Step Conditioned Deep Convolutional Neural Networks Improve Protein Secondary Structure Prediction. arXiv preprint arXiv:1702.03865v1, 2017.

25. Lin ZM, Lanchantin J, Qi YJ. MUST-CNN: a multilayer shift-and-stitch deep convolutional architecture for

sequence-based protein structure prediction. Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence 27-34, 2016.

26. Pollastri G, Przybylski D, Rost B, Baldi P. Improving the prediction of protein secondary structure in three and eight classes using recurrent neural networks and profiles. Proteins: Structure, Function, and Bioinformatics 47(2): 228-235, 2002.

27. Chou PY, Fasman GD. Prediction of protein conformation. Biochemistry 13(2): 222-245, 1974.

28. Jiang Q, Jin X, Lee SJ, Yao S. Protein secondary structure prediction: A survey of the state of the art. Journal of Molecular Graphics & Modelling 76: 379-402, 2017.

29. Wei Y, Xia W, Lin M, Huang J, Ni B, Dong J et al. HCP: A Flexible CNN Framework for Multi-label Image Classification. IEEE Transactions on Pattern Analysis & Machine Intelligence 38(9):1901-1907, 2016.

30. Cho K, Van Merriënboer B, Bahdanau D, Bengio Y. On the properties of neural machine translation: Encoder-decoder approaches. arXiv preprint arXiv:1409.1259, 2014.

31. Jozefowicz R, Zaremba W, Sutskever L. An empirical exploration of recurrent network architectures. International Conference on Machine Learning 2342-2350, 2015.

32. Kingma DP, Ba J. Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980, 2014.

33. Wang G, Dunbrack Jr RL. PISCES: a protein sequence culling server. Bioinformatics 19(12): 1589-1591, 2003.

34. Kryshtafovych A, Barbato A, Fidelis K, Monastyrskyy B, Schwede T, Tramontano A. Assessment of the assessment: evaluation of the model quality estimates in CASP10. Proteins: Structure, Function, and Bioinformatics 82(S2): 112-126, 2014.

35. Moult J, Fidelis K, Kryshtafovych A, Schwede T, Tramontano A. Critical assessment of methods of protein structure prediction (CASP)—round x. Proteins: Structure, Function, and Bioinformatics 82(S2): 1-6, 2014.

Yanbu Guo is currently a MS candidate in Yunnan University. His main research interests include deep learning and computational biology.

Bingyi Wang received his PhD in Ecology from Chinese Academy of Forestry. He received his MS in Botany from Yunnan University. His research interests include introduction, cultivation and molecular regulation in plants.

Weihua Li (Corresponding author**)** received her PhD in Information and Communication Engineering and MS in Computational Mathematics from Yunnan University. Her research interests include machine learning and bioinformatics.

Bei Yang revived her MS in Clinic Medicine from Kunming Medical University. Her research interest is in cardiology.