IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received October 13, 2020, accepted October 28, 2020, date of publication November 2, 2020, date of current version November 12, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3035208

Protein Secondary Structure Prediction Based on Generative Confrontation and Convolutional Neural Network

YAWU ZHAO, HUALAN ZHANG, AND YIHUI LIU
School of Computer Science and Technology, Qilu University of Technology (Shandong Academy of Sciences), Jinan 250353, China
Corresponding author: Yihui Liu (yxl@qlu.edu.cn)

This work was supported in part by the Natural Science Foundation of Shandong Province under Grant ZR2013FM020, and in part by the National Natural Science Foundation of China under Grant 61375013.

ABSTRACT In the field of bioinformatics, the prediction of protein secondary structure is a challenging task, and it is extremely important for determining the structure and function of proteins. In this paper, the generation of adversarial network and convolutional neural network model are combined for protein secondary structure prediction. Firstly, generate a confrontation network to extract protein features, and then combine the extracted features with the original PSSM data as the input of the convolutional neural network to obtain prediction results. Testsets CASP9, CASP10, CASP11, CASP12, CB513 and PDB25 obtained 87.06%, 87.24%, 87.31%, 87.39%, 88.13% and 88.93%, which are 3.88%, 4.6%, 7.97%,5.85%, 5.78%, 4.25% higher than one using the convolutional neural network alone. The experimental results show that the feature extraction ability of generating adversarial networks is very significant.

INDEX TERMS Bioinformatics, generative adversarial network, convolutional neural network, protein secondary structure prediction.

I. INTRODUCTION

Protein is one of the important biological macromolecules, which is indispensable to almost all life activities. With the completion of the Human Genome Project, scientists never stopped to study the structure of proteins. The classification information of protein structure should be solved in the field of protein research, and it is also very important in the field of bioinformatics [1]. Protein secondary structure prediction is a key step in tertiary structure prediction and it is a prerequisite for understanding and predicting tertiary structure. The improved accuracy of protein secondary structure prediction not only enables us to understand the complex relationship between protein sequence and protein structure, but also helps to analyze protein functions and manufacture drugs [2], so protein secondary structure prediction is a challenging task and of great significance. Using biological methods to determine the structure of proteins is expensive and time-consuming, therefore, we can predict the secondary structure of proteins with the help of computers.

The associate editor coordinating the review of this manuscript and approving it for publication was Vincenzo Conti.

In the field of bioinformatics, many computational methods have been used to predict the secondary structure of proteins. such as common machine learning algorithms including support vector machine [3], nearest neighbor algorithm [4] and Bayesian algorithm [5] etc. However, the feature extraction of machine learning depends on experience, which makes the feature extraction of data difficult. With the development of science and technology and the enhancement of computing power, people gradually pay attention to deep learning models [6]–[9]. It can learn features from original data without relying on expert experience. Protein secondary structure prediction was applied to convolutional neural networks (CNN) [10] and recurrent neural networks (RNN) [11] to improve prediction accuracy. The SPIDER3 [12] method utilized long-term and short-term memory bidirectional recurrent neural networks to capture longer amino acid sequence information, resulting in an accuracy rate of more than 80%. The SPOT-1D [13] method is the newer protein secondary structure prediction method at present, and it is an improvement of SPIDER3. On the basis of the SPIDER3 method, a residual convolutional network was combined to obtain better results. Ma *et al.* [14] proposed a method based on data segmentation and semi-random

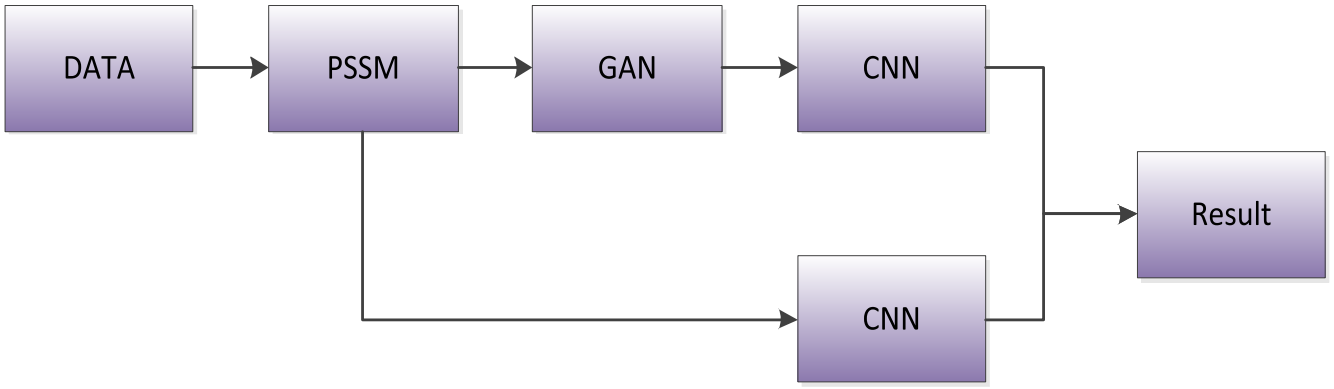

FIGURE 1. Model structure(GAN extracts features and combines with PSSM to form new features, and then uses CNN to predict Q_3 and SOV accuracy).

subspace. The accuracy of testing the 3-state on the 25PDB and CB513 datasets was 86.38% and 84.53%. The MUFOLD [15] method used a network named Deep3I, which is composed of two nested initial modules that can perform convolution operations, convolution and a fully connected dense layer, and effectively processing the local and global between amino acid residues interaction. Guo *et al.* [16] fused asymmetric convolutional neural networks and BiLSTM models, and performed eight class of protein secondary structure predictions. DeepCNF [17] combines deep neural networks and conditional neural fields to predict for 3- and 8-state secondary structure.

In recent years, generative adversarial network [18], [19], as a newer deep learning model, has significant effects in feature extraction and image denoising. Based on the above reasons, this paper integrates GAN and CNN neural networks, and proposes protein secondary structure prediction based on generative adversarial networks and convolutional neural networks. The generate adversarial network can extract the characteristics of amino acid residues through the game between the generator and discriminator, and the extracted features are fused with the original protein features and then send them to the convolutional neural network to predict protein secondary structure.

II. MODEL AND DATA

A. MODEL STRUCTURE

Protein secondary structure is based on protein sequence to predict the type of structure corresponding to amino acid residues, PSI-BLAST's position-specific scoring matrix (PSSM) [20] is used to represent protein sequences and contains abundant biological evolution information. The PSI-BLAST parameter is set to a threshold of 0.001 and 3 iterations to obtain a 20∗M PSSM matrix, where M is the length of the amino acid sequence and 20 represents the number of amino acid types. The definition of protein structure DSSP [21] contains eight structural types, namely H (α helix), B (β turn), E (fold), G (3-helix), I (5-helix), T (Corner), S (curl) and L (ring). In the experiment of this paper, G, H and I are replaced by H, B and E are replaced by E, and replace other structures with C.

TABLE 1. Number of proteins in the test dataset.

Test dataset	Number of proteins
CASP9	122
CASP10	99
CASP11	81
CASP12	19
CB513	513
PDB25	1672

In this paper, generative confrontation network and convolutional neural network were used to predict the secondary structure of proteins. Firstly, the data was preprocessed, and the PSSM matrix was divided according to the sliding windows of 13 and 19 to obtain the input data of the network. Its prediction model is shown in Figure 1.

B. DATASETS

In this paper, the ASTRAL [22] and CullPDB [23] datasets were used as the training set of the model. The CullPDB dataset was selected based on the percentage identity cutoff of 25%, the resolution cutoff of 3 angstroms, and the R-factor cutoff of 0.25. ASTRAL dataset had 6,892 proteins, with less than 25% sequence identity. We removed the protein with the same protein name in ASTRAL and CullPDB. There are a total of 15696 proteins. The test set used CASP [24]–[26] data set, including CASP9, CASP10, CASP11 and CASP12. In addition, the CB513 [27] and PDB25 [28] data sets are also used as the test set of the model, and the number of protein sequences of test sets is shown in Table 1.

III. MODEL PRINCIPLE AND RESULT EVALUATION

A. GENERATIVE ADVERSARIAL NETWORK

In 2014 Ian Goodfellow proposed Generative Adversarial Network [29], papers [18], [19] use generative adversarial networks for image denoising and feature extraction, which proves that generative adversarial networks have good characteristics. Generative adversarial network consists of two parts: generator and discriminator. The generator can learn the distribution characteristics of real data, in order to generate data similar to real protein data, while the discriminator is

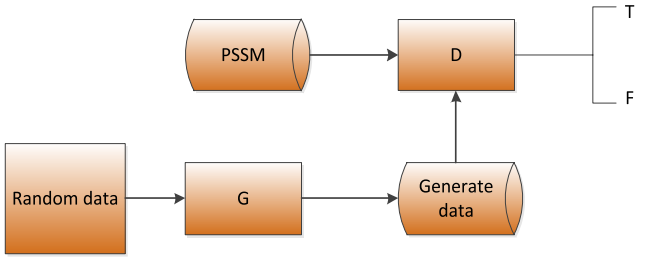

FIGURE 2. Generative adversarial network model.

to judge whether the data is generated by the generator or the real data, which is actually a binary classification problem. From the perspective of game theory, in order to improve the generating ability of the generator and the discriminating ability of the discriminator, they need to be optimized continuously, but eventually they reach the Nash equilibrium. The generator and discriminator can be represented by G and D respectively, and the generative adversarial network model is shown in Figure 2.

The learning process of GAN is the process of confrontation between D and G, D classifies the input protein matrix PSSM, D can discriminate the generated data from the real data, if the generated data is false, then D(G(z))=0, and if the real data is true, then $D(x) = 1$. When such a situation occurs, G needs to constantly adjust and optimize its parameters, so that the generated data is closer to the real data and D cannot judge whether the data is real or generated by G, namely D(G(z))=1. The process of confrontation between G and D is called a minimax game, its loss function is defined as follows.

$$\min_G \max_D V(D, G) = E_{x\sim p_{data}(x)}\left[\log D(x)\right] + E_{z\sim p_z(z)}\left[\log\left(1 - D\left(G\left(z\right)\right)\right)\right] \quad (1)$$

In the formula, x represents the real protein data, z represents the random data input to G, G (z) represents the fake data generated by the G network, and $D(x)$ represents the probability that the D network judges whether the real data is true. For D, the closer this value is to 1, the better. And D(G(z)) is the probability that the D network judges whether the protein data generated by G is true, the generator wants its own data to be closer to the real data, so G wants D(G(z)) to be as large as possible, at this time V(D,G) will become smaller, thus we see that the first symbol of equation (1) is min. The stronger the discriminator is, the bigger $D(x)$ should be, the smaller $D(G(x))$ should be, and then V(D,G) should be larger, so equation (1) is to find the maximum value for D.

In the generated confrontation model in this paper, the convolutional network is introduced into the G and D networks in order to improve the feature extraction capability of the generated confrontation network and to improve the prediction accuracy of protein secondary structure. The G network uses deconvolution for upsampling, and the activation function uses ReLU function. The D network uses a convolutional layer with a step size of 1, and the activation function uses a ReLU function. The features extracted from the generated adversarial network are combined with the PSSM matrix and the deep convolutional neural network is used to predict the secondary structure of the protein. The generator model is shown in Figure 3.

B. CONVOLUTIONAL NEURAL NETWORK

In recent years, as a popular deep learning algorithm, convolutional neural network has been applied to image processing [30], computer vision [31] and other fields. The method based on convolutional neural network [8], [9], [31] has been applied to protein secondary structure prediction, and has achieved remarkable results. Compared with the traditional neural network, it has the characteristics of weight sharing and local perception, which can reduce the network parameters and speed up the calculation. The structure diagram of the convolutional neural network model is shown in Figure 4.

The convolution layer performs feature extraction on the input protein data through the convolution kernel. The process of convolution is to perform an operation on the input protein matrix according to the size of the convolution kernel to generate a feature map with the same number of convolution kernels. The feature map is obtained by multiplying the input matrix and the weight and adding the offset, so that:

$$F_k^i = f\left(\sum_h P_h^{i-1} * W_k^i + b\right) \quad (2)$$

In the formula, f is the activation function ReLU, P_h^{i-1} represents the feature map obtained by the convolution kernel of the input data and the previous layer, W_k^i is a convolution kernel of the i-th layer, k represents the number of convolution kernels, i represents the number of convolution layers, b represents the offset parameter.

The pooling layer does not perform any learning and is often referred to as a nonlinear down-sampling form. The result of the pooling layer processing is to reduce the feature dimension and parameters to reduce the amount of calculation, increase the calculation speed, and also effectively reduce the overfitting. In addition, it has the characteristics of unchanged translation, which increases the robustness. In order to adjust the weights for training, the paper uses a back propagation algorithm using gradient descent algorithm.

The fully connected layer and the softmax layer are used as the output layer of the convolutional neural network. Each neuron of the fully connected layer must be connected to the neuron of the previous layer to output three types of protein secondary structures. The Softmax function layer uses the activation function to solve the classification problem of three types of protein structures. Its functions are defined as:

$$P(t_r/x) = \frac{P(x/t_r)\,P(t_r)}{\sum_{j=1}^{D} P\left(x/t_j\right)P\left(t_j\right)} = \frac{e^{o_r}}{\sum_{j=1}^{D} e^{o_j}} \quad (3)$$

where, $P(x/t_r)$ is the conditional probability of a given class sample, $P(t_r)$ is the prior probability of the protein structure

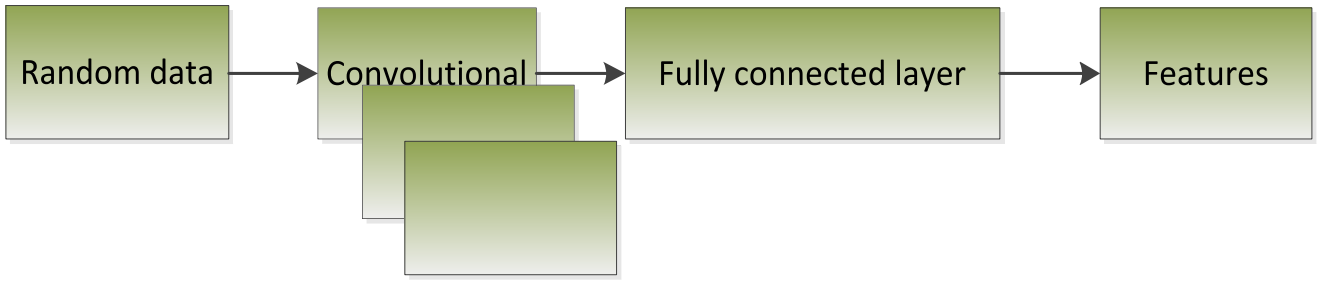

FIGURE 3. Generating an adversarial network generator model structure(The generator extracts features through a series of convolution operations.)

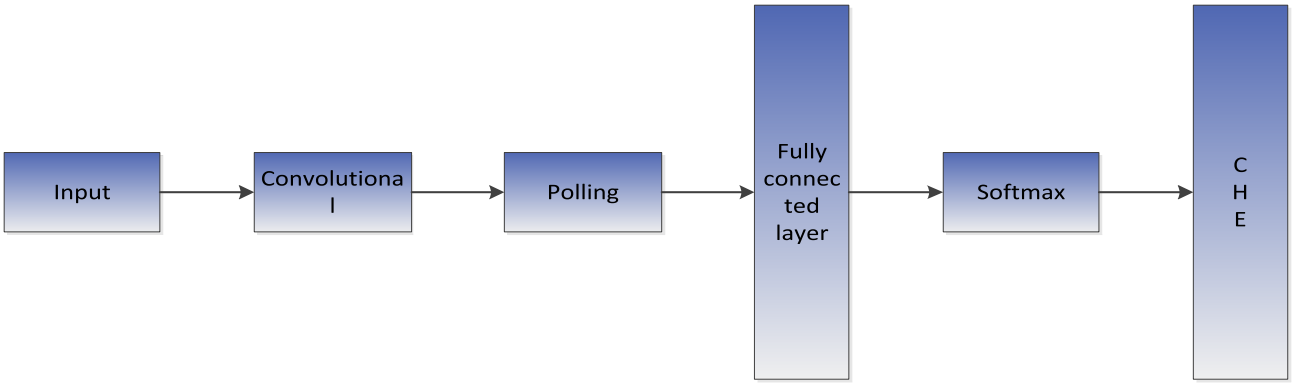

FIGURE 4. Convolutional neural network model(Input is a combination of GAN features and PSSM.)

class. The Softmax function is regarded as a multi-class extension of the logistic Sigmoid function [28].

C. RESULT EVALUATION

Generally, accuracy (Q_3) and segment overlap measure (SOV) [32] are widely used to evaluate the performance of protein secondary structure prediction.

According to DSSP [21] regulations, we convert G, H, I into H, E, B into E, and other structures into C. Then Q_3 represents the ratio of the number of correctly predicted amino acids in the three states to the entire amino acid sequence. The following formula is the definition of Q_3:

$$Q_3 = \frac{S_C + S_E + S_H}{S} \times 100\% \tag{4}$$

where S_C represents the number of accurately predicted protein structures of class C,S_E represents the number of accurately predicted protein structures of class E,S_H represents the number of accurately predicted protein structures of class H, S represents the total number of amino acids. The accuracy of each secondary structure can be calculated as:

$$Q_i = \frac{S_i}{S}, \quad i \in \{C, H, E\} \tag{5}$$

where Q_i idenotes the total number of the amino acid residues which are observed in the state i.

Sov is a measure based on the ratio of overlapping fragments. It is assumed that all observed structural fragments are marked as S_{ab}, and all predicted fragments are marked as S_{pr}, and S_a is a fragment with the same state of S_{ab} and S_{pr}. The length of any observed residue is defined as length(S_{ab}), for any pair of fragments in S_a, the actual length is minov (S_{ab}, S_{pr}), and the total length of at least one residue is maxov(S_{ab}, S_{pr}). Based on the above definition, the Sov calculation formula is as follows:

$$Sov = \frac{100}{n_{Sov}} \sum_{Sa} \left[\frac{\min ov\left(S_{ab}, S_{pr}\right) + \sigma\left(S_{ab}, S_{pr}\right)}{\max ov\left(S_{ab}, S_{pr}\right)} length\left(S_{ab}\right) \right] \tag{6}$$

Among them, the factor σ(Sab, Spr) is added, which allows the changes at the boundary of the observed fragment in the protein structure, which is defined as:

$$\sigma\left(S_{ab}, S_{pr}\right) = \min \begin{cases} \left(\max ov\left(S_{ab}, S_{pr}\right) - \min\left(S_{ab}, S_{pr}\right)\right) \\ \min ov\left(S_{ab}, S_{pr}\right) \\ \text{int}\left[len\left(S_{ab}\right)\right]/2 \\ \text{int}\left[len\left(S_{pr}\right)\right]/2 \end{cases} \tag{7}$$

N_{Sov} is the sum of the number of residues observed in all overlapping fragments in S_a plus the S_{ab} fragment that does not have the same predicted state.

IV. EXPERIMENTAL RESULTS

The experimental environment parameters of this paper are as follows: processor Intel(R) Xeon(R) Glod 5118 CPU

TABLE 2. Influence of the number of iterations on the accuracy using the length ofwindow 13.

Iterations	CASP9	CASP10	CASP11	CASP12	CB513	PDB25
10	84.4	84.71	84.25	85.04	86.23	86.66
20	84.52	85.26	84.09	84.33	86.24	86.82
30	82.76	85.27	84.58	85.01	86.21	86.72
40	84.4	85.46	84.13	84.33	86.19	86.84
50	84.38	84.60	84.09	83.81	86.30	85.52
60	84.29	84.90	84.36	84.05	86.15	86.80

TABLE 3. Influence of the number of iterations on the accuracy using the length ofwindow 19.

Iterations	CASP9	CASP10	CASP11	CASP12	CB513	PDB25
10	87.06	87.24	87.31	87.39	88.13	88.93
20	86.13	87.05	86.94	86.49	87.75	88.04
30	86.53	86.59	87.24	86.42	87.93	88.61
40	86.63	87.05	87.14	86.7	88.07	88.62
50	86.46	86.75	86.59	86.52	87.94	88.41
60	86.45	87.12	87.27	87.03	87.71	88.49

2.30GHz, graphics accelerator card RTX 2080Ti, operating system Linux, using Keras 2.3 version to build the model.

In order to evaluate the accuracy of the model in this article, six public test sets were used: CASP9, CASP10, CASP11, CASP12, CB513 and PDB25. In order to verify the effectiveness of generating adversarial networks, two different experiments were conducted to predict the secondary structure of three types of proteins. The first experiment is to use the convolutional neural network model for protein secondary structure prediction. The second experiment is to use the generated confrontation network to perform feature extraction on the protein data and then use the convolutional neural network to predict the protein secondary structure. In this paper, the length of sliding windows are 13 and 19, respectively, and the size and dimensions of the convolution kernel of the convolution layer are 11∗11∗270, 11∗11∗160 for the window length of 13 and 19∗19∗290, 16∗16∗170 for the window length of 19.

In order to verify the influence of the number of iterations on the feature extraction generated by the confrontation network, the unit of iteration times is ten thousand times. In this paper, the protein data under the 13 and 19 windows are verified, and the experimental results are shown in Table 2 and Table 3.

It can be seen from Table 2 and Table 3 that the accuracy rate is higher when the length of sliding window is 19, because more protein feature information can be contained using the window, length of 19 and with the increase of the number of iterations between the generator and the discriminator in the generation confrontation network, the accuracy rate shows a downward trend, and a better result is obtained when the number of iterations is 100,000. In the GAN training process, with the increase in the number of iterations, the discriminator can't distinguish the quality of the generated data, making the generated data and the original data very different. Therefore, the greater the number of iterations, the lower the Q_3 accuracy. In the generative confrontation network, in order to verify the effect of the number of layers, size and number of convolutions in the G network on the generated features, we adjust the hyperparameters. We test on the CASP10 data set, and the experimental results are shown in Table 4 Table 5 and Table 6. We use convolution kernel sizes of 3, 5, and 7, respectively, the number of convolution kernels are 128, 256, 512, and the number of convolution layers are 1, 2, and 3. For example, when the number of convolutional layers is 2, filter size is 3, the network structure is 3∗3, 3∗3.

In convolutional neural networks, in order to get the best network structure, we adjust the number of convolutional layers to get the best Q3 accuracy. Such as, when the number of convolutional layers is 2, the network structure is 19∗19, 19∗19, 16∗16, 16∗16. As can be seen in Table 2, when the number of network layers is 1, the Q3 accuracy is the highest.

We use convolutional neural network to predict protein structure, and the results are shown in Table 8.

By comparing Table 2, Table 3 and Table 8, it can be found that the features extracted from the generated adversarial network are fused with the PSSM matrix to predict the secondary structure of three types of proteins, and the Q_3 accuracy is greatly improved compared with the convolution neural network alone. It can be seen from the experiments in this paper that the feature extraction of the generated confrontation network is very effective, as shown in Table 9, which is improved by 3.88%, 4.6%, 7.97%, 5.85%,5.78% and 4.25% on the CASP9, CASP10, CASP11, CASP12, CB513, and PDB25 data sets, respectively. It proves the superiority of the feature extraction ability of generating adversarial networks.

The model in this paper is compared PSIPRED [33], RaptorX-SS8 [34] and DeepCNF [17], PSRM [14], MUFOLD-SS [15] models. As shown in Table 10, the

TABLE 4. The effect of the size and number of convolution kernels on the accuracy of Q_3.

Layers	Filter size	Filters	Q_3
1	3	128	85.96
		256	86.01
		512	86.24
	5	128	86.12
		256	86.78
		512	85.83
	7	128	85.46
		256	85.94
		512	86.53

TABLE 5. The effect of the size and number of convolution kernels on the accuracy of Q_3.

Layers	Filter size	Filters	Q_3
2	3	128	86..28
		256	86.54
		512	86.41
	5	128	87.02
		256	87.24
		512	87.13
	7	128	85.96
		256	86.23
		512	86.98

TABLE 6. The effect of the size and number of convolution kernels on the accuracy of Q_3.

Layers	Filter size	Filters	Q_3
3	3	128	86.33
		256	86.12
		512	86.64
	5	128	86.78
		256	87.17
		512	87.03
	7	128	86.87
		256	86.79
		512	86.65

TABLE 7. The influence of the number of convolutional layers on the accuracy of Q3.

Network layers	CASP9	CASP10	CASP11	CASP12	CB513	PDB25
1	87.06	87.24	87.31	87.39	88.13	88.93
2	87.01	87.16	86.99	87.23	88.04	88.53
3	86.98	87.03	86.49	87.05	88.10	88.74
4	86.75	86.95	86.89	86.31	87.62	87.52

TABLE 8. Convolutional neural network prediction accuracy.

Sliding window	CASP9	CASP10	CASP11	CASP12	CB513	PDB25
13	82.97	82.15	79.23	79.56	81.21	83.73
19	83.18	82.64	79.34	81.54	82.35	84.68

accuracy of predicting the secondary structure of three types of proteins was used as an index to evaluate the model in this paper. PSIPRED uses a two-layer feedforward neural network, RaptorX-SS8 uses a conditional neural field, and DeepCNF is a combination of a deep neural network and a conditional neural field. The results of PSIPRED,

TABLE 9. Q_3 and SOV accuracy in the datasets.

Dataset	Q_3	Q_C	Q_E	Q_H	SOV
CASP9	87.06	85.64	78.25	86.17	83.62
CASP10	87.24	79.45	76.18	89.31	81.98
CASP11	87.31	82.79	79.61	86.53	82.31
CASP12	87.39	81.15	78.55	90.78	80.03
CB513	88.13	84..39	77.73	87.37	84.26
PDB25	88.93	83.47	82.64	89.79	83.45

TABLE 10. Q_3 accuracy of the tested methods on 4 datasets.

Methods	CASP10	CASP11	CB513	CASP12
PSIPRED	81.2	80.7	79.2	-
RaptorX-SS8	78.9	79.1	78.3	-
DeepCNF	84.4	84.7	82.3	-
PSRM	85.51	85.89	84.83	85.55
MUFOLD-SS	85.98	85.39	-	80.5
Our Model	87.24	87.31	88.13	87.39

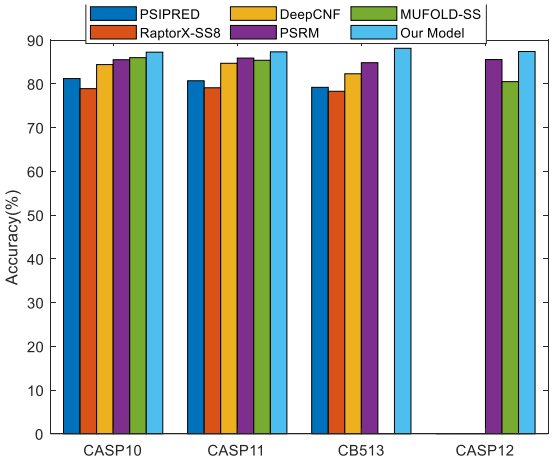

FIGURE 5. Q_3 Accuracy of our model and other 5 methods.

RaptorX-SS8, and DeepCNF on the test set are all taken from the literature [17].

V. CONCLUSION

As shown in Figure 5, compared with PSIPRED, RaptorX-SS8, DeepCNF, PSRM and MUFOLD-SS methods, our model has achieved better results on the CASP10 and CASP11 datasets. Experimental results show that our model is an effective method for predicting secondary structure.

Protein secondary structure prediction is a work of great significance in the field of bioinformatics, and is necessary to fully understand the function and structure of proteins. In this paper, the generative adversarial network and convolutional neural network model are combined to predict protein secondary structure. The generative adversarial network extracts the protein sequence features, and then uses the PSSM matrix as the input of the convolutional neural network to predict the protein secondary structure. Compared with the prediction results of only convolutional neural network, the feature extraction ability of the generated adversarial network is relatively strong, which can achieve very significant effects and has good scalability.

REFERENCES

[1] B. Zhang, J. Li, and Q. Lü, "Prediction of 8-state protein secondary structures by a novel deep learning architecture," *BMC Bioinf.*, vol. 19, no. 1, p. 293, Dec. 2018.

[2] Y. Yang, J. Gao, J. Wang, R. Heffernan, J. Hanson, K. Paliwal, and Y. Zhou, "Sixty-five years of the long march in protein secondary structure prediction: The final stretch?" *Briefings Bioinf.*, vol. 19, no. 3, pp. 482–494, 2018.

[3] Y. Wang, J. Cheng, Y. Liu, and Y. Chen, "Prediction of protein secondary structure using support vector machine with PSSM profiles," in *Proc. IEEE Inf. Technol., Netw., Electron. Autom. Control Conf.*, May 2016, pp. 502–505.

[4] R. Bondugula, O. Duzlevski, and D. Xu, "Profiles and fuzzy K-nearest neighbor algorithm for protein secondary structure prediction," in *Proc. 3rd Asia–Pacific Bioinf. Conf.*, Jan. 2005, pp. 85–94.

[5] V. Robles, P. Larrañaga, J. M. Peña, E. Menasalvas, M. S. Pérez, V. Herves, and A. Wasilewska, "Bayesian network multi-classifiers for protein secondary structure prediction," *Artif. Intell. Med.*, vol. 31, no. 2, pp. 117–136, Jun. 2004.

[6] R. Heffernan, K. Paliwal, J. Lyons, J. Singh, Y. Yang, and Y. Zhou, "Single-sequence-based prediction of protein secondary structures and solvent accessibility by deep whole-sequence learning," *J. Comput. Chem.*, vol. 39, no. 26, pp. 2210–2216, Oct. 2018.

[7] M. S. Klausen, "Netsurfp-2.0: Improved prediction of protein structural features by integrated deep learning," *BioRxiv*, vol. 87, no. 6, pp. 520–527, 2018, Art. no. 311209.

[8] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," *Commun. ACM*, vol. 60, no. 6, pp. 84–90, May 2017.

[9] Z. Li and Y. Yu, "Protein secondary structure prediction using cascaded convolutional and recurrent neural networks," in *Proc. Int. Joint Conf. Artif. Intell. (IJCAI)*, 2016, pp. 1–8.

[10] Y. Liu and J. Cheng, "Protein secondary structure prediction based on wavelets and 2D convolutional neural network," in *Proc. 7th Int. Conf. Comput. Syst.-Biol. Bioinf. (CSBio)*, 2016, pp. 53–57.

[11] J. Hanson, K. Paliwal, T. Litfin, Y. Yang, and Y. Zhou, "Improving prediction of protein secondary structure, backbone angles, solvent accessibility and contact numbers by using predicted contact maps and an ensemble of recurrent and residual convolutional neural networks," *Bioinformatics*, vol. 35, no. 14, pp. 2403–2410, Jul. 2019, doi: 10.1093/bioinformatics/bty1006.

[12] R. Heffernan, Y. Yang, K. Paliwal, and Y. Zhou, "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*, vol. 33, no. 18, pp. 2842–2849, Sep. 2017.

[13] J. Hanson, K. Paliwal, T. Litfin, Y. Yang, and Y. Zhou, "Improving prediction of protein secondary structure, backbone angles, solvent accessibility and contact numbers by using predicted contact maps and an ensemble of recurrent and residual convolutional neural networks," *Bioinformatics*, vol. 35, no. 14, pp. 2403–2410, Jul. 2019.

[14] Y. Ma, Y. Liu, and J. Cheng, "Protein secondary structure prediction based on data partition and semi-random subspace method," *Sci. Rep.*, vol. 8, no. 1, p. 9856, Dec. 2018.

[15] C. Fang, Y. Shang, and D. Xu, "MUFOLD-SS: New deep inception-inside-inception networks for protein secondary structure prediction," *Proteins, Struct., Function, Bioinf.*, vol. 86, no. 5, pp. 592–598, May 2018.

[16] Y. Guo, W. Li, B. Wang, H. Liu, and D. Zhou, "DeepACLSTM: Deep asymmetric convolutional long short-term memory neural models for protein secondary structure prediction," *BMC Bioinf.*, vol. 20, no. 1, p. 341, Dec. 2019, doi: 10.1186/s12859-019-2940-0.

[17] S. Wang, J. Peng, J. Ma, and J. Xu, "Protein secondary structure prediction using deep convolutional neural fields," *Sci. Rep.*, vol. 6, no. 1, pp. 1–11, May 2016.

[18] R. Wang, X. Xiao, B. Guo, Q. Qin, and R. Chen, "An effective image denoising method for UAV images via improved generative adversarial networks," *Sensors*, vol. 18, no. 7, p. 1985, Jun. 2018.

[19] S. Yu, H. Chen, E. B. G. Reyes, and N. Poh, "GaitGAN: Invariant gait feature extraction using generative adversarial networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jul. 2017.
[20] D. T. Jones, "Protein secondary structure prediction based on position-specific scoring matrices 1 1Edited by G. Von Heijne," *J. Mol. Biol.*, vol. 292, no. 2, pp. 195–202, Sep. 1999.
[21] W. Kabsch and C. Sander, "Dictionary of protein secondary structure: Pattern recognition of hydrogen-bonded and geometrical features," *Biopolymers*, vol. 22, no. 12, pp. 2577–2637, Dec. 1983.
[22] N. K. Fox, S. E. Brenner, and J.-M. Chandonia, "SCOPe: Structural classification of proteins—Extended, integrating SCOP and ASTRAL data and classification of new structures," *Nucleic Acids Res.*, vol. 42, no. D1, pp. D304–D309, Jan. 2014.
[23] G. Wang and R. L. Dunbrack, "PISCES: Recent improvements to a PDB sequence culling server," *Nucleic Acids Res.*, vol. 33, no. Web Server, pp. W94–W98, Jul. 2005.
[24] J. Moult, K. Fidelis, A. Kryshtafovych, and A. Tramontano, "Critical assessment of methods of protein structure prediction (CASP)-round IX," *Proteins, Struct., Function, Bioinf.*, vol. 79, no. S10, pp. 1–5, 2011.
[25] J. Moult, K. Fidelis, A. Kryshtafovych, T. Schwede, and A. Tramontano, "Critical assessment of methods of protein structure prediction (CASP)—Round x," *Proteins, Struct. Funct. Bioinform.*, vol. 82, pp. 1–6, Feb. 2014.
[26] J. Moult *et al.*, "Critical assessment of methods of protein structure prediction (CASP)—Round X," *Proteins, Struct. Funct. Bioinf.*, vol. 79, pp. 1–5, Oct. 2012.
[27] J. A. Cuff and G. J. Barton, "Evaluation and improvement of multiple sequence methods for protein secondary structure prediction," *Proteins, Struct., Function, Genet.*, vol. 34, no. 4, pp. 508–519, Mar. 1999.
[28] K. D. Kedarisetti, L. Kurgan, and S. Dick, "Classifier ensembles for protein structural class prediction with varying homology," *Biochem. Biophys. Res. Commun.*, vol. 348, no. 3, pp. 981–988, Sep. 2006.
[29] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Proc. Int. Conf. Neural Inf. Process. Syst.* Cambridge, MA, USA: MIT Press, 2014, pp. 2672–2680.
[30] G. Chartrand, P. M. Cheng, E. Vorontsov, M. Drozdzal, S. Turcotte, C. J. Pal, S. Kadoury, and A. Tang, "Deep learning: A primer for radiologists," *RadioGraphics*, vol. 37, no. 7, pp. 2113–2131, Nov. 2017.
[31] J. Zhou, H. Wang, Z. Zhao, R. Xu, and Q. Lu, "CNNH_PSS: Protein 8-class secondary structure prediction by convolutional neural network with highway," *BMC Bioinf.*, vol. 19, no. S4, p. 60, May 2018.
[32] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," *Commun. ACM*, vol. 60, no. 6, pp. 84–90, May 2017.
[33] L. J. McGuffin, K. Bryson, and D. T. Jones, "The PSIPRED protein structure prediction server," *Bioinformatics*, vol. 16, no. 4, pp. 404–405, Apr. 2000, doi: 10.1093/bioinformatics/16.4.404.
[34] S. Wang, W. Li, S. Liu, and J. Xu, "RaptorX-property: A Web server for protein structure property prediction," *Nucleic Acids Res.*, vol. 44, no. W1, pp. W430–W435, Jul. 2016, doi: 10.1093/nar/gkw306.

YAWU ZHAO is currently pursuing the Master of Engineering degree in computer application technology with the Qilu University of Technology (Shandong Academy of Sciences). His research interests include deep learning, intelligent information processing, and biomedical information processing.

HUALAN ZHANG received the bachelor's degree in computer science and technology from the Qilu University of Technology (Shandong Academy of Sciences). Her research interest includes bioinformatics.

YIHUI LIU received the Ph.D. degree in computer science from the University of Nottingham, U.K., in 2004. She is currently a Professor with the Qilu University of Technology (Shandong Academy of Sciences), China. She has published more than 40 articles. Her research interests include biomedical information and medical image analysis, including protein structure prediction, microarray data analysis, and so on.