Article

A Bi-LSTM Based Ensemble Algorithm for Prediction of Protein Secondary Structure

Hailong Hu ^{1,2}, Zhong Li ^{1,3,*}, Arne Elofsson ⁴ and Shangxin Xie ³¹ Faculty of Mechanical Engineering & Automation, Zhejiang Sci-Tech University, Hangzhou 310018, China² School of Science, Zhejiang A&F University, Hangzhou 311300, China³ School of Science, Zhejiang Sci-Tech University, Hangzhou 310018, China⁴ Department of Biochemistry and Biophysics, Science for Life Laboratory, Stockholm University, Stockholm, 17121 Solna, Sweden

* Correspondence: lizhong@zstu.edu.cn; Tel.: +86-571-8684-3540

Received: 31 July 2019; Accepted: 20 August 2019; Published: 28 August 2019

Abstract: The prediction of protein secondary structure continues to be an active area of research in bioinformatics. In this paper, a Bi-LSTM based ensemble model is developed for the prediction of protein secondary structure. The ensemble model with dual loss function consists of five sub-models, which are finally joined by a Bi-LSTM layer. In contrast to existing ensemble methods, which generally train each sub-model and then join them as a whole, this ensemble model and sub-models can be trained simultaneously and the performance of each model can be observed and compared during the training process. Three independent test sets (e.g., data1199, 513 protein Cuff & Barton set (CB513) and 203 proteins from Critical Appraisals Skills Programme (CASP203)) are employed to test the method. On average, the ensemble model achieved 84.3% in Q₃ accuracy and 81.9% in segment overlap measure (SOV) score by using 10-fold cross validation. There is an improvement of up to 1% over some state-of-the-art prediction methods of protein secondary structure.

Keywords: protein secondary structure; sequence analysis; Bi-LSTM; ensemble algorithm; deep learning

1. Introduction

Currently, the sustained progress of high-throughput genome sequencing methods is providing an exponentially increasing amount of known protein sequences. However, it is practically impossible to do detailed experimental studies for all proteins due to the high cost and low efficiency. As a result, an urgent requirement is to use amino acid sequences to predict the protein structure and function. One important task in such pipelines is to predict the protein secondary structure.

The major methods for predicting protein secondary structure can broadly be classified into two categories: Template-based methods and sequence profile-based methods. For the template-based methods, statistical models are frequently used to analyze the probability of specific amino acids appearing in different secondary structure elements [1,2]. Generally, researchers have to construct the structural template database from known protein structures with certain sequence similarity, and then find alignments between the whole query sequence or its short fragments and sequences in the protein structure template database. The prediction of these methods is ideal only in the case that sequences similar to the query sequence can be found in the template database [3]. Sequence profile based methods not only make use of the sequence profile information, but also benefit from the structure information. The sequence profile typically represented as position specific scoring matrix (PSSM) [4] is constructed based on the multiple sequence alignment between the query sequence and similar sequences. Sequence profile-based methods perform well in the case when a good PSSM is built, since some similar sequences to the query sequence exist in the template database. In other cases,

it is difficult to obtain successful results with these methods [5]. As the prediction of protein secondary structure is important for predicting the 3D structure and function of proteins, it is still an active field in bioinformatics and other related fields. The accuracy of three-state prediction increases gradually from <70% to 82%–84% [6–8].

In recent decades, many machine learning methods, especially support vector machine (SVM) [9], random forest classifier and Markov model [10], have been utilized in the prediction of protein secondary structure. There are some drawbacks with such methods, for example, they cannot deal with sequences with varied lengths which often exist in the training data and cannot capture the long-range dependence among the same protein sequences. In order to solve these problems, various neural network [11] have recently been employed to predict the protein structure. We know that the local contexts, specifically, the neighbors of each amino acid, are critical for the prediction of protein secondary structure. However, the long-range interactions, referring to amino acid residues that are far from each other in their sequence positions but are close in the three-dimensional space, are also vital for the prediction. The Long Short-Term Memory (LSTM) cell proposed by Hochreiter and Schmidhuber [12] has the ability to learn both distant and close intra-sequence dependencies. It has been used in many artificial intelligence tasks and has achieved great success in fields such as speech recognition [13], natural language processing [14], and bioinformatics [15].

In this paper, we apply a Bi-LSTM-based ensemble model for the prediction of protein secondary structure. Various classification rules for protein secondary structure may somehow impact the accuracy of the prediction. Based on the Define Secondary Structure of Proteins (DSSP) method [16], each amino acid residue is assigned to one of the three states: H (α -Helix), E (β -strands), C (random coil). Generally, DSSP provides an eight state assignment of secondary structure denoted by single letter codes: H, T, S, I, G, E, B and C. These states are converted into three classes using the following convention: {H}->H, {E}->E, {B, C, G, I, T, S}->C. The main advantages of our work are as follows: (a) Five type features of protein are used to fully explore the properties of protein sequence; (b) a BI-LSTM based ensemble model consists of five sub-models is proposed; (c) dual loss functions are employed to the ensemble model. These attributions make the ensemble method achieve satisfactory prediction results for protein secondary structure.

2. Materials and Performance Measure

2.1. Datasets

Our training set contains non-homologous protein sequences extracted from the Protein Data Bank (PDB) database (<http://www.rcsb.org/pdb/home/home.do>). To guarantee the generality of our model, it is different from conventional methods whose training sets are generally 25% sequence similarity cutoff. The redundancy of our training data is first reduced at the 25% sequence similarity level, and then further reduced by using a sensitive search tool lightning-fast iterative protein sequence searching by HMM-HMM alignment (HHBlits) to exclude all proteins that have a weak hit ($E < 0.1$), resulting in 5530 protein sequences. The initial size of the training set was 7720 protein sequences. The training set excluded proteins with chain breaks and those shorter than 35 residues.

A 10-fold cross validation was applied for the dataset to train the ensemble model. The ratios of the three states (α -Helix, β -sheet, coil) of amino acid residues in the training set are 33.7%, 27.6% and 38.7%, respectively. We used three publicly available datasets (513 protein Cuff & Barton set (CB513), data1199 and 203 proteins from Critical Appraisals Skills Programme (CASP203)) to evaluate the performance of our method. The CB513 dataset [17], containing 513 non-homologous protein sequences, is often used as the benchmark dataset to evaluate the performance of prediction methods about protein structures. The second test set came from Ref [18], including 1199 non-redundant sequences; data1199 hereafter. The third test set was derived from the 2016 CASP meeting, containing 203 non-redundant proteins, named CASP 203. To ensure the validity of the test result, no two sequences in the training and test sets have a similarity over 30%.

2.2. Performance Measure

Overall, accuracy (Q_3) [19], Matthews correlation coefficients (MCC) [20], and segment overlap measure (SOV) [21] are widely used to evaluate the prediction performance of protein secondary structure. For the individual amino acid, Q_3 accuracy and MCC are fairly good evaluation indexes. Generally, several adjacent amino acids compose the α -helices and β -strands structures. Even a high accuracy of the individual amino acid does not mean that the prediction of protein secondary structure is truly correct. Therefore, we adopted the SOV index, which is a more appropriate measure for the prediction accuracy of protein secondary structure.

(a) Q_3 accuracy

The confusion matrix has broadly been applied to measure the performance of classifiers. A confusion matrix used to evaluate the prediction accuracy of protein secondary structure was first proposed in Ref [17]. It defined M with a size of 3×3 , the number of residues observed in the state i but predicted as the state j was denoted as M_{ij} , where $i, j \in \{H, E, C\}$. The following formula is the definition of Q_3 :

$$Q_3 = \frac{1}{N} \sum_{i=1}^3 M_{ii} \quad (1)$$

where N is the total number of the residues in the dataset. The accuracy of each secondary structure can be calculated as

$$Q_i = \frac{1}{n_i} M_{ii} \quad (2)$$

where n_i in the denominator denotes the total number of the amino acid residues which are observed in the state i .

(b) Matthews correlation coefficient (MCC)

The Matthews correlation coefficient is used in machine learning as a measure for the quality of binary classifications, and also to evaluate the prediction accuracy of protein secondary structure. Specifically, for a particular state $i \in \{H, E, C\}$, the corresponding MCC is given by

$$MCC_i = \frac{p_i \times n_i - u_i \times o_i}{\sqrt{(p_i + u_i) \times (p_i + o_i) \times (n_i + u_i) \times (n_i + o_i)}} \quad (3)$$

where $p_i = M_{ii}$, $n_i = \sum_{j \neq i} \sum_{k \neq i} M_{jk}$, $o_i = \sum_{j \neq i} M_{ji}$, $u_i = \sum_{j \neq i} M_{ij}$.

(c) The segment overlap measure (SOV)

Differing from Q_3 and MCC, SOV represents the true prediction performance by calculating the average overlap between the observed and predicted states. For a particular state $i \in \{H, E, C\}$, it is defined as

$$SOV_i = \frac{1}{n_i} \sum_{S_1} \frac{MINOV(S_1, S_2) + DELTA(S_1, S_2)}{MAXOV(S_1, S_2)} \quad (4)$$

where S_1 and S_2 denote the observed and predicted results for the state $i \in \{C, E, H\}$, S_i is the number of all overlapping segment pairs (S_1, S_2) in the state i ; $MINOV(S_1, S_2)$ represents the length of the actual overlap between S_1 and S_2 ; $MAXOV(S_1, S_2)$ is the length of the total extent for which either the segments S_1 or S_2 has a residue in the state i ; n_i denotes the total number of amino acids residues that are observed in the state i . The definition of $DELTA(S_1; S_2)$ is

$$DELTA(S_1, S_2) = \min \left\{ \begin{array}{c} MAXOV(S_1; S_2) - MINOV(S_1; S_2) \\ MINOV(S_1; S_2) \\ INT(0.5 \times LEN(S_1)) \\ INT(0.5 \times LEN(S_2)) \end{array} \right\} \quad (5)$$

where $LEN(S_1)$ denotes the number of amino acid residues in segment S_1 and INT is the integral function.

SOV is an overall index for three states and is calculated as

$$SOV = \frac{1}{N} \sum_{i \in \{H,E,C\}} \sum_{S(i)} \left(\frac{MINOV(S_1, S_2) + DELTA(S_1, S_2)}{MAXOV(S_1, S_2)} \times LEN(S_1) \right) \quad (6)$$

where $S(i)$ is the number of all overlapping segment pairs (S_1, S_2) in the state i .

3. Features and Methods

3.1. Features Selection

In this work, five types of protein feature were employed for the prediction of protein secondary structure. (1) Seven representative physio-chemical properties of amino acids can directly be obtained from R [22], including Hydrophobicity, Grapn shape index, polarizability, normalized van der Waals volume, random coil Ca chemical shifts, localized electrical effect and PK (RCOOH). (2) The 20-dimensional position specific substitution matrices (PSSM) scores can be acquired by applying PSI-BLAST [4] to search the uniprot_sprot database (ftp://ftp.uniprot.org/pub/databases/uniprot/current_release/knowledgebase/complete) to generate sequence profiles (with three iterations and E-values set to 0.001). (3) The 20-dimensional PSSM count also comes from PSI-BLAST [4] (with three iterations and E-values set to 0.001). PSSM count means the number of substitutions separately for each position in a protein multiple sequence alignment. PSSM scores are a function of PSSM count, which can be positive or negative integers. Positive scores indicate that the given amino acid substitution occurs more frequently in the alignment than expected by chance, while negative scores indicate that the substitution occurs less frequently than expected. (4) The 30-dimensional hidden Markov model (HMM) sequence profiles can be determined by applying HHBlits [23] to search the uniprot20_2016_02 database (http://wwwuser.gwdg.de/~compbiol/data/hhsuite/databases/hhsuite_dbs/old-releases/). (5) A 1-dimensional positive integer is obtained by the word embedding [24], which encodes the type of each amino acid residue of proteins to generate more abstract and dense features in the new feature space. In this paper, 50-dimensional dense features were generated and put into the neuron network. These features describe protein sequences according to individual amino acid property, protein homology and mathematical encoding, respectively.

3.2. The Ensemble Algorithm Based on Bi-LSTM

Compared to methods using a single model, ensemble methods have gradually been used for various tasks and achieve satisfactory performance in bioinformatics. For example, Ref [25] proposed an ensemble method that combines four predictors and uses a random forest classifier for the prediction of human sub-cellular localization. Four predictors, based either on support vector machine [26] or naive Bayes [27], employed different features of protein to make predictions.

Inspired by their work, we propose an ensemble algorithm for the sequence-based prediction of protein secondary structure. Five types of feature were extracted to represent a protein sequence, and these features describe protein sequences according to individual amino acid property, protein homology and mathematical encoding. Naturally, five sub-models were used to compose the ensemble model and one type of feature is put into one of the five sub-models. As shown in Figure 1, five sub-models (pssm_model, hmm_model, pssm_count_model, pps_model, wordem_model) compose the ensemble model. Each sub-model contains two bi-directional LSTM layers, followed by two more fully connected layers with 512 and 128 neurons, respectively. The fully-connected neurons in each network use rectified linear unit (ReLU) as activations. Note that since the sizes of these five types of features are different among the five sub-models, the numbers of neurons in bi-directional LSTM layers are different. There are 100, 128, 100, 64 and 128 neurons per direction in bi-directional LSTM layers

for the pssm_model, hmm_model, pssm_count_model, pps_model and wordem_model, respectively. Besides, the embedding size of wordem_model was set as 50 [24]. These sub-models were joined by a bi-directional LSTM layer with 64 neurons per direction followed by two fully connected layers with 128 and 64 neurons, respectively, to form the ensemble model. In Figure 1, there are 7, 20, 20, 30 and 50 features fed as the input for the pps_model, pssm_model, pssm_count_model, hmm_model, and wordem_model. There were 640 (128×5) features finally fed the input for the final Bi-LSTM layer. Note that each number in Figure 1 denotes the neuron number for that layer. To reduce the network over-fitting, we utilize the dropout trick [28] with a dropout ratio of 70% in fully connected layers during the training. As the ensemble model has many neurons, we need a higher dropout ratio to prevent over-fitting. Adam optimization [29] with the initial learning rate of 0.0005 was used to train our model, which is an efficient stochastic optimization able to compute the adaptive learning rates during the training process. A 10-fold cross validation and early stopping was applied during the training to prevent from the over-fitting. Note that the early stopping was only applied to the ensemble model, so when the ensemble model stopped training, all sub-models stopped training too. The clip gradients trick [30] was also applied to prevent gradients from being exploded, which usually occurs in the LSTM-based method. Additionally, we used two Tesla K80 GPUs with 64G memory to speed up the training. The number of neurons of each sub-model was selected through the different experiments. When choosing these settings, we found that we could obtain satisfying prediction results by our ensemble method. As mentioned above, one type of feature is entered into one of the five sub-models and the outputs from the last fully connected layer in the network of each sub-model are combined by a bi-directional LSTM layer. A soft-max function is applied to the last fully connected layer of both sub-models and ensemble model to get normalized probabilities. Note that this ensemble model contains dual loss functions; specifically, one for the sub-models and another for the ensemble model. All parameters in each sub-model would be updated during both back-propagation processes from sub-loss and global loss, but parameters in the last bi-directional LSTM layer and in the last two fully connected layers are updated only during the back-propagation process from global loss. In our method, while the ensemble model is trained once, each sub-model is trained twice.

Figure 1. The ensemble model based on Bi-LSTM.

Compared to traditional ensemble methods, which typically train each sub-model consecutively, our ensemble model possesses some advantages. For example, (1) the ensemble model and the five sub-models can be trained at the same time; (2) the performance of each model can be observed and

compared during the training process; (3) five types of protein feature are terminally fused by the final bi-directional LSTM layer, which can improve the performance of our method.

4. Results and Discussion

We first compared the results by our ensemble model with those by sub-models. As shown in Figure 2, Q_3 accuracy and sub-class accuracies from a 10-fold cross validation by different models were represented by different colors. As each sub-model and the ensemble model was trained in parallel in our method, the training time of our model was not much different from that of other methods.

Figure 2. Q_3 accuracy of ensemble model and five sub-models for the validation set.

The Q_3 accuracy for the validation set was 84.37% with corresponding Q_H , Q_E and Q_C accuracies of 88.31%, 80.35% and 81.65%, respectively. Among the five sub-models, the results of model-hmm were the best, with the Q_3 accuracy of 82.26% and corresponding Q_H , Q_E and Q_C accuracies of 86.67%, 78.34% and 81.27%, respectively. Compared to model-hmm, the ensemble model achieved an improvement in the Q_3 accuracy of 2.11%. From Figure 2, we can refer that the Q_3 accuracy and the corresponding Q_H , Q_E and Q_C accuracies for the validation set of the ensemble model were higher than those of any sub-models.

Table 1 reports the SOV scores and MCC measurements for the validation set by the ensemble model and the five sub-models. The SOV score of the ensemble model was 81.92% and the corresponding SOV_H , SOV_E and SOV_C scores were 86.47%, 82.29% and 77.38%, respectively. Among the sub-models, hmm_model performed the best, achieving an SOV score of 80.1% and corresponding SOV_H , SOV_E and SOV_C scores of 83.07%, 79.83% and 75.51%, respectively. Compared to the hmm_model, the ensemble model achieved an improvement of 1.9% in SOV and 3.38%, 2.45% and 1.87% in SOV_H , SOV_E and SOV_C , respectively. The three MCC measurements (C_E , C_H , C_C) by the ensemble model were higher than those of the five sub-models.

Table 1. Segment Overlap Measure (SOV) scores and (matthews correlation coefficients) MCC results by six different methods for the validation set.

Methods	Accuracy Measures						
	SOV (%)	SOV _H (%)	SOV _E (%)	SOV _C (%)	C _H	C _E	C _C
ensemble_model	81.917	86.468	82.289	77.382	0.838	0.769	0.697
pssm_model	74.482	76.246	70.825	72.112	0.761	0.693	0.642
hmm_model	80.138	83.086	79.831	75.512	0.796	0.782	0.682
pssm_c_model	64.831	71.456	58.314	61.087	0.656	0.589	0.537
pps_model	64.821	68.257	56.752	65.251	0.648	0.572	0.542
wordem_model	64.785	66.721	56.543	63.563	0.653	0.558	0.562

From the results in Figure 2 and Table 1, it can be seen that the performance of different sub-models is quite unstable. Some sub-models have higher accuracy in helical prediction, while others have higher accuracy in coil or sheet prediction. It seems that the evaluation indexes by pssm_model and pssm_c_model are somewhat similar, and those by pps_model and wordem_model are also similar, but their performance is quite different at the single protein level. Specifically, for a given protein, the error made by different sub-models will overlap to some extent, but they are never identical. Even in the worst sub-model and the best sub-model, the correct predictions of the former are never totally included in that of the latter. In other words, the five sub-models are complementary to each other and the ensemble model takes the advantage of each sub-model to improve the effectiveness of our method. We find that pssm_model and hmm_model outperform others because their features are more effective. We tried to use pssm_model and hmm_model to combine an ensemble model before, but the classification accuracy for the same data is about 0.8% lower than that of the current ensemble model. We also see that the SOV measure is the lowest for coil residues, which may be caused by the dependency on protein structural classification. Although 3_{10} -helices and b-bridges constitute short secondary structure segments that have some structural similarity to alpha-helix and beta-strand, they are classified as coil residues. Generally, prediction methods are more precise in the core of regular secondary structure segments than at the termini.

In order to demonstrate the effectiveness of our ensemble model, we implemented our method to predict protein secondary structure for three independent test sets, data1199, CB513 and CASP203. We selected three widely used methods, single-sequence-based prediction method employing LSTM-BRNNs (SPIDER3) [18], JPred4 [31] and RaptorX [32] for the comparison with our method. These three methods used an iterative deep learning neural network, JNet algorithm, and deep convolutional neural fields Deep Convolutional Neural Fields (DeepCNF) to predict protein secondary structure. As mentioned above, the similarity between protein sequences in the test and training sets was quite low, to prevent overestimating the performance of our predictor.

The Q_3 accuracy and the corresponding Q_H , Q_E and Q_C accuracies of secondary structure prediction at individual amino acid obtained from data1199 by the above mentioned methods are shown in Figure 3. The Q_3 accuracy of our ensemble model is 84.0% with corresponding Q_H , Q_E and Q_C accuracies of 86.7%, 79.2% and 84.1%, respectively. Compared to SPIDER3, JPred4 and RaptorX, our method achieved an improvement in the Q_3 accuracy of 0.7%, 4.7% and 2.5%, respectively.

Figure 3. Q_3 Accuracy of ensemble model and the other three methods for data1199.

Table 2 reports the SOV scores and MCC measurements of secondary structure prediction at individual amino acid from data1199 by different methods. The SOV score of the ensemble model was 81.6% and SOV_H , SOV_E and SOV_C scores were 85.7%, 84.4% and 76.6%, respectively. Compared to SPIDER3, JPred4 and RaptorX, our method achieved an improvement in the SOV scores of 0.7%, 10.6%

and 5.0%, respectively, though it was 1.8% lower than SPIDER3 in SOV_H . Compared to other methods, it achieved some improvements in C_H , C_E and C_C .

Table 2. SOV scores and MCC results by four different methods for data1199.

Methods	Accuracy Measures						
	SOV (%)	SOV _H (%)	SOV _E (%)	SOV _C (%)	C _H	C _E	C _C
ensemble	81.56	85.69	84.38	76.57	0.8059	0.7917	0.6862
SPIDER3	80.90	87.50	81.40	73.70	0.7905	0.7518	0.6524
JPred4	70.90	77.90	74.20	64.10	0.7236	0.6771	0.5891
RaptorX	76.50	81.00	80.40	71.20	0.7833	0.7375	0.6515

The Q_3 accuracy and the corresponding Q_H , Q_E and Q_C accuracies of secondary structure prediction at individual amino acid obtained from CB513 by above mentioned methods are shown in Figure 4. The Q_3 accuracy of the ensemble model was 83.5% with corresponding Q_H , Q_E and Q_C accuracies of 85.5%, 80.1% and 82.6%, respectively. Compared to SPIDER3, JPred4 and RaptorX, our method achieved an improvement in the Q_3 accuracy of 0.6%, 6.5% and 1.1%, respectively.

Figure 4. Q_3 accuracy of ensemble model and other three methods for 513 protein Cuff & Barton set (CB513).

Table 3 reports the SOV scores and MCC measurements of secondary structure prediction at individual amino acid from CB513 by different methods. The SOV score of our method was 80.5%, and SOV_H , SOV_E and SOV_C scores were 84.7%, 83.2% and 75.8%, respectively. Compared to SPIDER3, JPred4 and RaptorX, our method achieved an improvement in the SOV scores of 2.6%, 9.6% and 4.0%, respectively. The three MCC measurements of our method for the test set were higher than those by other methods.

Table 3. SOV scores and MCC results by four different methods for 513 protein Cuff & Barton set (CB513).

Methods	Accuracy Measures						
	SOV (%)	SOV _H (%)	SOV _E (%)	SOV _C (%)	C _H	C _E	C _C
ensemble	80.50	84.70	83.20	75.80	0.8052	0.7964	0.6942
SPIDER3	77.90	83.50	79.40	72.70	0.7905	0.7518	0.6524
JPred4	70.90	77.90	74.20	64.10	0.7236	0.6771	0.6291
RaptorX	76.50	81.00	80.40	71.20	0.7833	0.7675	0.6735

In order to test the validity of our method, we constructed another test set based on the CASP database by applying PSI-BLAST. It contained 203 proteins and the criterion of selection was that we randomly chose the proteins in CASP and guaranteed those proteins are non-homologous to our training set. We initially constructed the training set from the PDB database with non-homologous protein sequences. Here, we constructed the test set from the CASP database with non-homologous protein sequences to furthermore verify our method for identifying the protein structure from sequences. As shown in Figure 5, Q_3 value of our method was 83.3% and higher than those of SPIDER3, JPred4 and RaptorX, which were 81.9%, 79.3% and 81.0% respectively. Table 4 reported the other indexes showing that the SOV result of our ensemble model outperformed the SPIDER3, JPred4 and RaptorX results. For C_H , C_E and C_C , we found that most results of our method were higher than those of SPIDER3, JPred4 and RaptorX. Overall, our method was more accurate in helix and sheet prediction than coil prediction.

Figure 5. Q_3 accuracy of ensemble model and other three methods for 203 proteins from Critical Appraisals Skills Programme (CASP203).

Table 4. SOV scores and MCC results by four different methods for 203 proteins from Critical Appraisals Skills Programme (CASP203).

Methods	Accuracy Measures						
	SOV (%)	SOV _H (%)	SOV _E (%)	SOV _C (%)	C_H	C_E	C_C
ensemble	80.60	84.50	82.60	75.45	0.8058	0.7854	0.6977
SPIDER3	78.10	83.20	79.20	72.50	0.7978	0.7126	0.6721
JPred4	72.20	76.80	75.10	64.60	0.7324	0.6796	0.6043
RaptorX	75.40	80.80	82.30	71.20	0.7839	0.7598	0.6757

5. Conclusions

In this paper, we introduce a Bi-LSTM-based ensemble method for the prediction of protein secondary structure. Experimental results show that our method is often better than other well-known methods. The method is available as a prediction sever at <http://ipv.math.sci.zstu.edu.cn/proteinPrediction.jsp> and the example code is also available at this website. Our algorithm can also be applied to other fields of bioinformatics, such as the prediction of the relative solvent accessibility and disordered regions (DISO), thus expanding its potential. Additionally, we currently predict three states of protein secondary structure. We can similarly extend it to predict eight states of protein secondary structure, which will be our future work.

Hanson et al. [33] used PSSM and HMM as input features and trained many distinct models based on Bi-LSTM and ResNets. They put all features into each model and selected the nine best models after

reviewing their performance on a validation set to ensemble the final model. Unlike their work, our ensemble model lets one type of features be put into one of the five sub-models to fully explore the properties of the protein sequence. Additionally, the ensemble model and the five sub-models can be trained at the same time.

Although we obtain satisfying prediction results for most protein sequences, the performance of the ensemble model is degenerated for extremely long protein sequences, for example, when the number of residues in a protein sequence exceeds 1000. More powerful architecture with an attention mechanism, such as neural Turing machines [34], may be suitable for solving this problem. In addition, combining profiles or structural similarity with machine learning methods would be a referable strategy for further improving the performance.

Author Contributions: Conceptualization, H.H., S.X. and Z.L.; methodology, H.H. and Z.L.; software, S.X.; validation, H.H., S.X. and Z.L.; formal analysis, A.E.; writing—original draft preparation, H.H. and S.X.; writing—review and editing, Z.L. and A.E.; funding acquisition, Z.L. and H.H.

Funding: This work was supported by National Natural Science Foundation of China under Grant No. 11671009, Zhejiang Provincial Natural Science Foundation of China under Grant No. LZ19A010002 and LY19F010014.

Acknowledgments: We would like to thank National Natural Science Foundation and Zhejiang Provincial Natural Science Foundation of China for the research support.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

LSTM	Long Short-Term Memory
PSSM	Position specific scoring matrix
Q ₃	Overall accuracy
SOV	Segment overlap measure
MCC	Matthews correlation coefficient
PSI-BLAST	Position-Specific Iterative BLAST

References

1. Ward, J.J.; McGuffin, L.J.; Buxton, B.F.; Jones, D.T. Secondary structure prediction with support vector machines. *Bioinformatics* **2003**, *19*, 1650–1655. [[CrossRef](#)]
2. Xie, S.X.; Li, Z.; Hu, H.H. Protein secondary structure prediction based on the fuzzy support vector machine with the hyperplane optimization. *Gene* **2018**, *642*, 74–83. [[CrossRef](#)] [[PubMed](#)]
3. Bondugula, R.; Xu, D. MUPRED: A tool for bridging the gap between template based methods and sequence profile based methods for protein secondary structure prediction. *Proteins Struct. Funct. Bioinf.* **2010**, *66*, 664–670. [[CrossRef](#)] [[PubMed](#)]
4. Altschul, S.F.; Madden, T.L.; Schäffer, A.A.; Zhang, J.; Zhang, Z.; Miller, W.; Lipman, D.J. Gapped BLAST and PSI-BLAST: A new generation of protein database search programs. *Nucleic Acids Res.* **1997**, *25*, 3389–3402. [[CrossRef](#)] [[PubMed](#)]
5. Geourjon, C.; Deléage, G. SOPM: A self-optimized method for protein secondary structure prediction. *Protein Eng. Des. Sel.* **1994**, *7*, 157–164. [[CrossRef](#)]
6. Rost, B. Review: Protein secondary structure prediction continues to rise. *J. Struct. Biol.* **2001**, *134*, 204–218. [[CrossRef](#)] [[PubMed](#)]
7. Yaseen, A.; Li, Y. Context-based features enhance protein secondary structure prediction accuracy. *J. Chem. Inf. Model.* **2014**, *54*, 992–1002. [[CrossRef](#)] [[PubMed](#)]
8. Wang, S.; Peng, J.; Ma, J.; Xu, J. Protein secondary structure prediction using deep convolutional neural fields. *Sci. Rep.* **2016**, *6*, 18962. [[CrossRef](#)] [[PubMed](#)]
9. Cortes, C.; Vapnik, V. Support-vector networks. *Mach. Learn.* **1995**, *20*, 273–297. [[CrossRef](#)]
10. Karplus, K. SAM-T08, HMM-based protein structure prediction. *Nucleic Acids Res.* **2009**, *37*, 492–497. [[CrossRef](#)]
11. Heffernan, R.; Paliwal, K.; Lyon, J.; Dehzangi, A.; Sharma, A.; Wang, J.; Sattar, A.; Yang, Y.; Zhou, Y. Improving prediction of secondary structure, local backbone angles, and solvent accessible surface area of proteins by iterative deep learning. *Sci. Rep.* **2015**, *5*, 11476. [[CrossRef](#)] [[PubMed](#)]

12. Hochreiter, S.; Schmidhuber, J. Long short-term memory. *Neural Comput.* **1997**, *9*, 1735–1780. [[CrossRef](#)] [[PubMed](#)]
13. Hinton, G.; Deng, L.; Yu, D.; George, E.; Mohamed, D.A.; Jaitly, N.; Senior, A.; Vanhoucke, V.; Nguyen, P.; Sainath, T.N.; et al. Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups. *IEEE Signal Process. Mag.* **2012**, *29*, 82–97. [[CrossRef](#)]
14. Gal, Y.; Ghahramani, Z. A theoretically grounded application of dropout in recurrent neural networks. *Adv. Neural Inf. Process. Syst.* **2016**, 285–290.
15. Hanson, J.; Yang, Y.; Paliwal, K.; Zhou, Y. Improving protein disorder prediction by deep bidirectional long short-term memory recurrent neural networks. *Bioinformatics* **2017**, *33*, 685–692. [[CrossRef](#)] [[PubMed](#)]
16. Kabsch, W.; Sander, C. Dictionary of protein secondary structure: Pattern recognition of hydrogen-bonded and geometrical features. *Biopolym. Orig. Res. Biomol.* **1983**, *22*, 2577–2637. [[CrossRef](#)] [[PubMed](#)]
17. Rost, B.; Sander, C. Prediction of protein secondary structure at better than 70% accuracy. *J. Mol. Biol.* **1993**, *232*, 584–599. [[CrossRef](#)] [[PubMed](#)]
18. 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* **2017**, *33*, 2842–2849. [[CrossRef](#)]
19. Clementi, C.; García, A.E.; Onuchic, J.N. Interplay among tertiary contacts, secondary structure formation and side-chain packing in the protein folding mechanism: All-atom representation study of protein L. *J. Mol. Biol.* **2003**, *326*, 933–954. [[CrossRef](#)]
20. Matthews, B.W. Comparison of the predicted and observed secondary structure of T4 phage lysozyme. *Biochim. Biophys. Acta* **1975**, *405*, 442–451. [[CrossRef](#)]
21. Zemla, A.; Venclovas, C.; Fidelis, K.; Rost, B. A modified definition of Sov, a segment-based measure for protein secondary structure prediction assessment. *Proteins Struct. Funct. Bioinf.* **1999**, *34*, 220–223. [[CrossRef](#)]
22. Fauchère, J.L.; Charton, M.; Kier, L.B.; Verloop, A.; Pliska, V. Amino acid side chain parameters for correlation studies in biology and pharmacology. *Int. J. Pept. Protein Res.* **2010**, *32*, 269–278. [[CrossRef](#)]
23. Remmert, M.; Biegert, A.; Hauser, A.; Söding, J. HHblits: Lightning-fast iterative protein sequence searching by HMM-HMM alignment. *Nat. Methods* **2012**, *9*, 173–175. [[CrossRef](#)]
24. Levy, O.; Goldberg, Y. Neural word embedding as implicit matrix factorization. *Adv. Neural Inf. Process. Syst.* **2014**, *3*, 2177–2185.
25. Salvatore, M.; Warholm, P.; Shu, N.; Basile, W.; Elofsson, A. SubCons: A new ensemble method for improved human subcellular localization predictions. *Bioinformatics* **2017**, *33*, 2464–2470. [[CrossRef](#)] [[PubMed](#)]
26. Li, Z.; Wang, J.; Zhang, S.; Zhang, Q.; Wu, W. A new hybrid coding for protein secondary structure prediction based on primary structure similarity. *Gene* **2017**, *618*, 8–13. [[CrossRef](#)] [[PubMed](#)]
27. Murakami, Y.; Mizuguchi, K. Applying the Naïve Bayes classifier with kernel density estimation to the prediction of protein–protein interaction sites. *Bioinformatics* **2010**, *26*, 1841–1848. [[CrossRef](#)] [[PubMed](#)]
28. Srivastava, N.; Hinton, G.; Krizhevsky, A.; Sutskever, I.; Salakhutdinov, R. Dropout: A simple way to prevent neural networks from overfitting. *J. Mach. Learn. Res.* **2014**, *15*, 1929–1958.
29. Kingma, D.; Ba, J. Adam: A method for stochastic optimization. *Comput. Sci.* **2014**, arXiv:1412.698012.
30. Pascanu, R.; Mikolov, T.; Bengio, Y. On the difficulty of training recurrent neural networks. *Int. Conf. Mach. Learn.* **2013**, 1310–1318.
31. Drozdetskiy, A.; Cole, C.; Procter, J.; Barton, G.J. JPred4: A protein secondary structure prediction server. *Nucleic Acids Res.* **2015**, *43*, 389–394. [[CrossRef](#)] [[PubMed](#)]
32. Wang, S.; Li, W.; Liu, S.; Xu, J. RaptorX-Property: A web server for protein structure property prediction. *Nucleic Acids Res.* **2016**, *44*, 430–435. [[CrossRef](#)] [[PubMed](#)]
33. Hanson, J.; Paliwal, K.; Litfin, T.; Yang, Y.; Zhou, Y. 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* **2018**, *35*, 2403–2410. [[CrossRef](#)] [[PubMed](#)]
34. Bahdanau, D.; Cho, K.; Bengio, Y. Neural machine translation by jointly learning to align and translate. *Comput. Sci.* **2014**, arXiv:1409.04739.