Bagging MSA Learning: Enhancing Low-quality PSSM with Deep Learning for Accurate Protein Structure Property Prediction

Yuzhi Guo, Jiaxiang Wu, Hehuan Ma, Sheng Wang, and Junzhou Huang*

University of Texas at Arlington, Arlington, TX, 76019, USA
Tencent AI Lab, Shenzhen, 518057, China

Abstract. Accurate predictions of protein structure properties, *e.g.* secondary structure and solvent accessibility, are essential in analyzing the structure and function of a protein. PSSM (Position-Specific Scoring Matrix) features are widely used in the structure property prediction. However, some proteins may have low-quality PSSM features due to insufficient homologous sequences, leading to limited prediction accuracy. To address this limitation, we propose an enhancing scheme for PSSM features. We introduce the "Bagging MSA" method to calculate PSSM features used to train our model, and adopt a convolutional network to capture local context features and bidirectional-LSTM for long-term dependencies, and integrate them under an unsupervised framework. Structure property prediction models are then built upon such enhanced PSSM features for more accurate predictions. Empirical evaluation of CB513, CASP11, and CASP12 datasets indicate that our unsupervised enhancing scheme indeed generates more informative PSSM features for structure property prediction.

Keywords: Deep learning· Unsupervised learning · Enhancing PSSM · Protein secondary structure prediction.

1 Introduction

The function of a protein is closely related to its structure, which is largely determined by the amino-acid sequence. However, predicting one protein's structure based on its amino-acid sequence alone remains an open and challenging problem. An alternative approach is to firstly predict structure properties, including secondary structure, solvent accessibility, and backbone dihedral angles [1]. Those predictions are combined eventually to help the final prediction of protein structure.

PSSM (Position-Specific Scoring Matrix) features [3], which reflect per-residue evolution patterns in the sequence profile, are commonly used in the structure property prediction [4, 5]. The quality of PSSM features is basically determined by the underlying multiple sequence alignments (MSA) [6]. MSA requires searching the query amino-acid sequence through a large-scale sequence database, *e.g.* UniRef [20] and UniClust [21]. The MSA quality of the protein can be

*: Corresponding: jzhuang@uta.edu

evaluated by counting the number of homologous proteins, or the non-redundant sequence homologs(Meff [2]) retrieved from the database. However, for those proteins with a limited number of high-quality homologous sequences, the prediction quality is often limited due to less informative PSSM features [7]. One possible solution is to develop more efficient and accurate MSA search algorithm, such as SABERTOOTH [8], hhblits [9], jackhmmer [10], and HBLAST [11]. These algorithms have achieved certain performance improvement by speeding up the searching process, as well as find more accurate homologous protein sequences in the database. However, if the database did not contain enough homologous protein sequences for the target protein, it is still inaccessible to obtain sufficient quantity or high quality of the MSA, yet the corresponding high-quality PSSM features.

In this paper, we propose an unsupervised deep learning method to enhance the low-quality PSSM features of proteins. To be specific, during the training of our model, we randomly sample the MSA of each protein in a certain proportion in each learning iteration, which we called "Bagging MSA". Then, we use the "Weak PSSMs" calculated by these bags and the "Original PSSM" calculated by all MSA to train our network. In this way, our network can learn how to generate high-quality PSSM from a protein that has low-quality PSSM features.

The most commonly predicted one-dimensional structural property of a protein is the secondary structure. Therefore, in order to evaluate our method on different prediction networks, we use two widely used deep learning techniques in the protein secondary prediction area, which are CNN and bi-LSTM models [28, 35, 29]. The knowledge of the secondary structure of proteins and the network of validation of our method are described in section 2 and section 3.

The technical contributions of this paper are summarized as: 1) Our method is the first attempt to enhance low quality PSSMs of proteins. According to the experimental results, our method significantly improve the secondary structure prediction task of proteins with weak PSSM. 2) In the unsupervised module, our method calculate PSSM features by randomly sampling 10% to 20% MSA in each training iteration as the input data, and use the original PSSM features as unsupervised labels. This approach not only increases the diversity of the data, but also make the network more flexible to learn different PSSM quality differences so as to give full play to unsupervised learning. 3) Our method is generalizable since it is capable for any prediction model with PSSM as the input other than just secondary protein prediction task. 4) The unsupervised part of our method is independent, so the output could be used as the input directly for the inference phase of any prediction network, which is more flexible and efficient.

2 Related work

2.1 Position-Specific Scoring Matrix

MSA A multiple sequence alignment (MSA) is a sequence alignment of multiple homologous protein sequences for the target protein[12]. See Fig. 1 for an

example of MSA. MSA is an important step in comparative analyses and property predicting of biological sequences, since a lot of information *e.g.* evolution and co-evolution clusters, are displayed on the MSA and can be mapped to the target sequence of choice or on the protein structure [13]. Almost all existing approaches to studying proteins utilize MSAs indirectly, that is, they convert MSAs into a position-specific scoring matrix (PSSM) that represents the distribution of amino acid types on each column [14].

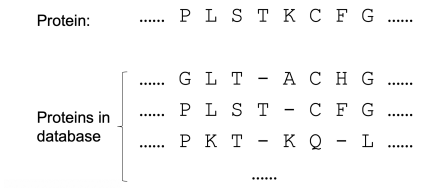

Fig. 1. An example of MSA.

PSSMs calculation PSSM scores are generally expressed as positive or negative integers. A positive score indicates that the frequency of substitutions in a given amino acid sequence is higher than expected, while a negative score indicates that the frequency of substitutions is lower than expected [15, 16].

We extract the PSSM features of size $n \times 21$ based on Eq.(1) and Eq.(2), where, n is the protein sequence length, 21 is the sum of twenty known amino acids appeared in the genetic code and one unknown amino acid marker. *Frequency* is the count of occurrences of residue j (j = 1,2,3, ..., 21) in column i (i = 1,2,3, ..., n), 20 represents the known amino acids. A simple procedure called pseudo-counts assigns minimal scores to residues which do not appear at a certain position of the alignment according to the following equation(1), where we set the *Pseudocount* equal to 1. N is the number of sequences in the multiple alignments. The *Background frequency* in Eq.(2) is the frequency of each residue appearing in the entire MSA of the protein.

$$score_{i,j} = \frac{Frequency + Pseudocount}{N + 20(Pseudocount)} \tag{1}$$

$$PSSM_{i,j} = \log(score/Backgroundfrequency) \tag{2}$$

2.2 Scoring criteria for PSSM

Count score The number of sequence homologs is recorded as the Count score. As we mentioned before, PSSM is a matrix calculated from the MSA, and the quality of the MSA directly determines the quality of the PSSM. We can use the number of homologous proteins of the MSA to evaluate the quality of the PSSM, which is represented as Count score. The larger Count score leads to more reliable PSSM. Thus, the Count score is one important criteria to evaluate the quality of the PSSM features.

Meff score We introduce the Meff score as the number of non-redundant sequence homologs. As in [7], homologous sequence in MSA of proteins have some redundancy, so we use Meff score as another criteria for PSSM to demonstrates the superiority and stability of our model under various evaluation standards.

The calculation formula of Meff score is shown in Eq.(3). where both i and j go over all the sequence homologs, $S_{i,j}$ is a binary number which describes the similarity of two proteins. We use the hamming distance to compute the similarity of two sequence homologs[17]: $S_{i,j}$ is 1 if the normalized hamming distance is less than 0.3; otherwise $S_{i,j}$ is set to 0.

$$Meff = \sum_i \frac{1}{\sum_j S_{i,j}} \tag{3}$$

2.3 Protein secondary structure prediction

The sequence space of proteins is vast, with perhaps 20 residues at each position, and evolution has been sampling it over billions of years. One of the most important sub-problems in protein studies is the secondary structure prediction. Protein secondary structure refers to the local conformation of the polypeptide backbone of proteins. There are two regular SS states: alpha-helix (H) and beta-strand (E), and one irregular SS type: coil region (C) [18]. The other way is a DSSP algorithm [19] to classify SS into 8 fine-grained states. In particular, the algorithm assigns 3 types for helix (G, H and I), 2 types for strand (E and B), and 3 types for coil (T, S and L). Overall, many computational methods have been developed to predict both 3-state secondary structure and a few to predict 8-state secondary structure. Meanwhile, since a chain of 8-state secondary structures contains more precise structural information for a variety of applications [25, 37], the focus of secondary structure prediction has been shifted from 3-state secondary structure(Q3) prediction to the prediction of 8-state secondary structures(Q8). Because the Q8 problem is much more complicated than the Q3 problem, deep learning methods would be more suitable for addressing the Q8 problem.

3 Method

3.1 Framework overview

Our method consists of two stages: enhancing PSSM and secondary structure prediction. The workflow of the inference phase is shown in Fig. 2. We input the low-quality PSSM into the trained unsupervised model with the protein sequence features to generate enhanced PSSM features. Then the enhanced PSSM features with sequence features are concatenated as the input of the inference phase for the prediction network. Finally, the results of the enhanced PSSM and the original PSSM on the prediction model are compared for evaluation.

3.2 Unsupervised Learning to enhance PSSM

The architecture of our unsupervised learning method is shown in Fig. 3, which mainly contains four parts: Bagging MSA module, Local contexts feature encoding module, Long-distance interdependencies feature encoding module and Generation module. For each amino acid in a protein sequence, its input features are concatenated by its sequence features and PSSM features, which form a $2l$ (l=21) dimensional vector. We denote the size of the entire input features as $N \times 2l$, and the size of the output from unsupervised learning network is $N \times l$, where N is the length of the protein sequence. The details regarding input features are explained in the experiments section.

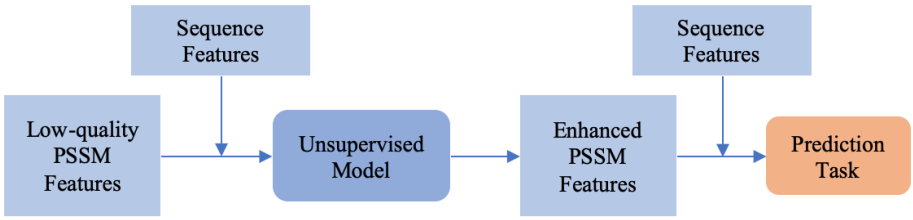

Fig. 2. Framework Overview.

Bagging MSA The main purpose of our enhancing PSSM module is to generate higher-quality PSSM features from low-quality PSSM features calculated from MSA with fewer rows or lower quality. Here we introduce the concept of 'Bagging MSA': As shown in Fig. 3, we randomly sample a small part of MSA for a protein and repeat this operation in each training iteration and for each protein. We bring in a hyper-parameter R to determine the proportion of selected homologous proteins in MSA randomly per training iteration, *e.g.* when R = [10%, 20%], a number greater than 10% and less than 20% would be randomly selected for each batch, and the homologous proteins in MSA would be randomly sampled according to this proportion. In this way, we are able to get many MSA bags, and each MSA bag would calculate a so-called 'Weak PSSM'. We used the weak PSSM calculated by these bags as a part of the input unsupervised data, and the original PSSM calculated by the complete MSA as the unsupervised labels. This module is ideal for unsupervised learning due to the size of the PSSM matrix is always the same for the same protein, even though the MSA size of each bag and label is different.

Local contexts feature encoding module We introduce a fully convolutional architecture as the local contexts feature encoding module. Recently, CNN has been successfully used in the seq2seq model [23] and machine translation [24], as well as applied in several protein studies, which achieved remarkable successes [25, 26]. This one-dimensional convolution operation is usually used to process sequence data, such as emotional analysis and sequence structure prediction [27, 28], so CNN would be a good fit for our prediction task.

In our method, the local contexts feature encoding module exploits the One-dimensional convolution to extract the local hidden patterns and features of adjacent amino-acid residues from the input matrix. This module contains three 1-d convolutional layers with the ReLU activation function, and the window size is equal to three for each layer, as shown in Appendix A.1.

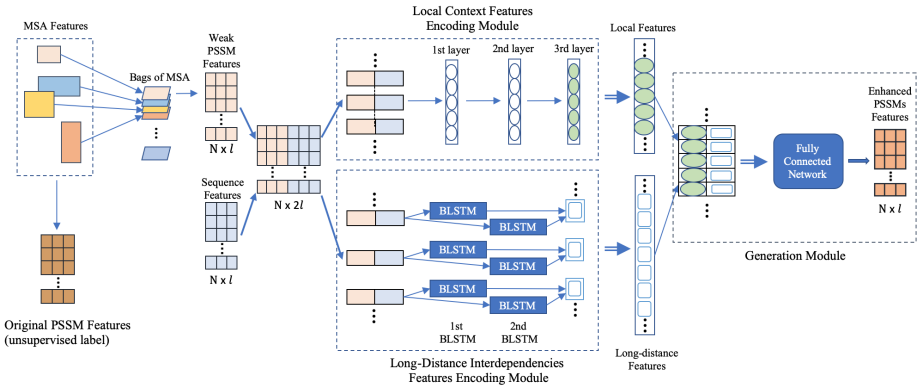

Fig. 3. Unsupervised learning model. 1) Bagging MSA Module has two outputs:"Original PSSM" calculated by all MSA are used as the unsupervised labels; "Weak PSSM" calculated via the bags of MSA are fed into the two encoding networks. 2) The outputs of the two encoding networks are local features and long-distance features respectively. 3) The output of the generation module is the "Enhanced PSSM", which is used to calculate the loss from the "Original PSSM" to adjust the networks.

Long-distance interdependencies feature encoding module As we mentioned before, CNNs have the ability to capture local relationships of spatial or temporal structures, but we can not capture sufficient long-range sequence information by increasing the window size and network depth infinitely. However, long-distance interdependencies [29] of amino-acid residues are also critical for amino acid sequence information. Inspired by the success of some methods which use a combination of multiple neural networks, for example, coupling residual two-dimensional bidirectional long short-term memory with convolutional neural networks [30], ACLSTM[31] and CRRNNs [32], our method not only uses convolutional neural network with a few layers but also another network to catch Long-distance interdependencies feature.

RNN-based model has achieved remarkable results in sequence modeling; however, the gradient vector may grow or degrade exponentially over a long sequence during the training process. Thus LSTM neural networks are designed to avoid this problem by introducing the gate structures, which is good at capturing the long-range relations (from the first atom to the last one).

In our method, the long-distance interdependencies feature encoding module includes two stacked bidirectional LSTM neural networks. As shown in Appendix A.1, the input data are fed into the feature encoding model by its original order as well as the reverse order, and then the two outputs are concatenated together as the final features representation.

Generation module Our method has one fully connected hidden layer in the generation module. Moreover, in order to get the complete information of protein sequence, as shown in Fig. 3, we directly concatenate the outputs of the previous two modules and feed them into the fully connected(FC) layer with the ReLU activation function to generate the enhanced PSSMs. We use the MSE loss[22] to adjust our unsupervised network, as shown in Eq.4.

$$Loss_{unsup} = MSE(PSSM_{Enhanced}, PSSM_{Full}) \tag{4}$$

3.3 Prediction network

Since our unsupervised learning method is an independent enhancing PSSM network, we are able to use any deep learning network for the prediction module to verify the generalization of our method. In this study, we use two protein secondary structure prediction networks to evaluate our method: CNN-based network and LSTM-based network, which are two widely used deep learning prediction networks. For CNN-based method, we use five CNN layers [28], and fix the window size to 11 since the average length of an alpha-helix is around eleven residues [33] and that of a beta-strand is around six [34]. For LSTM-based method, we use two stacked bidirectional LSTM neural networks [35] and a fully connected(FC) layer.

The input data for the prediction network is the same as the input for the unsupervised learning model, which is the concatenation of sequence information and PSSM features calculated by the complete MSA of the protein. The protein secondary structure is used as the label. Based on the validation results, we select the best model as the secondary structure predictor, then feed the enhanced PSSM features generated by our unsupervised network and the original PSSM into the predictor respectively. Last, the prediction performances of the two PSSM features are compared to evaluate the effectiveness of our enhanced PSSM model.

4 Experiments

4.1 Experiments set up

Dataset We use four publicly available datasets: CullPDB [36] of 5926 proteins, CB513 [37] of 513 proteins, CASP11 of 85 proteins, and CASP12 of 40 proteins. CASP11 and CASP12 datasets are downloaded from the official CASP website [38]. 53 duplicated proteins observed in the CullPDB are removed and 591 proteins are randomly sampled for validation, then the remaining proteins are used for training. The other three datasets are used as the test dataset. We generate the position specific scoring matrix (PSSM) by searching the Uniref50 [40] database. And the labels used for the prediction network are 8-state protein secondary structures which are generated by DSSP [19, 44].

Input features The input features for the encoding networks of our method are described in [37]. We extract the MSA from Uniref50 databases using Jackhmmer [10], and set the parameters refer to their guide [41], details are listed in Appendix A.3. We randomly sample 10% to 20%(R = [10%, 20%]) of the MSA for each protein within each learning iteration(Bagging MSA), and then we calculate PSSM using Eq.(1) and Eq.(2). We transform those PSSMs by the Sigmoid function 1/(1+exp(-x)) where x is a PSSM entry to map each PSSM value in between 0 and 1. As shown in Fig. 3, the input features of the two encoding modules is a $N \times 2l$ matrix, where N is the length of the input sequence and $2l$ is the dimension of the concatenated vectors. In our method, the sequence feature vectors are sparse one-hot vectors of 21 elements(l=21) since there might be some unknown amino acids in a protein sequence. Therefore, there are 42 input features in total for each residue, 21 from PSSM features and the other 21 from sequence feature.

For the prediction part, there are 42 input features for each residue too, 21 of them are from PSSM features and the others are from sequence feature. We compare the testing results of the enhanced input features with the original input features to evaluate the effectiveness of our unsupervised model.

Neural network structure and learning Hyper-parameters The framework of our unsupervised learning method is very flexible in the network structure selection.

In the long-distance interdependencies feature encoding module, we can set different hidden layers and hidden dimensions (with different layers and layer hidden sizes). Moreover, different types of network can be chosen in addition to the bi-LSTM network, such as LSTM [42]. Due to the space limitation of this paper, two stacked bi-LSTM with 512 hidden units are used for all experiments. Then, we use 1d-CNN of 3 hidden layers, and 100 neurons for each layer in the local contexts feature encoding module. The window size at each layer is set to 3.

For optimization, we use multi-step LR(learning rate) descent with [30,100,200] for epoch indices. The multiplicative factor of learning rate decay is 0.1. We use Adam [43] as the optimizer of our method. The initial learning rate for all training models is 0.0001.

For the protein secondary structure prediction task, we have two kinds of networks. For CNN network, we use five 1-dim CNN layers with window size 11, and neurons size 100 for each layer. For LSTM network, we use two stacked bi-LSTM with 512 hidden units and one fully connected(FC) layer.

Evaluation metric For the unsupervised learning, we calculate the RMSE of the Enhanced PSSM and the Original PSSM in the input feature as the evaluation matrix. Q8 accuracy is the criterion of the prediction module.

4.2 Results

Relationship between PSSM quality and performance As we mentioned before, we use two methods to score the quality of the protein PSSM, higher score represents better quality. Fig. 4 and Fig. 5 show the relationship between the quality of PSSM and the corresponding performance on the prediction networks on CB513 dataset. Fig. 4 shows the average accuracy obtained by using Count score as the evaluation standard on the prediction network of CNN and LSTM respectively, and Fig. 5 for the Meff score. We can find that proteins with high-quality PSSM performs better than proteins with low-quality PSSM both CNN-based and LSTM-based prediction network, as well as under all evaluations including Count score or Meff score. Table 1 and table 2 show the data distribution within the ranges Count and Meff Scores. Thus, our method aims at improving the prediction performance for those proteins with original low-quality PSSM by enhancing their PSSM features. See the gray-scale images in Appendix A.2, which show the difference between "before" and "after" PSSM enhancement.

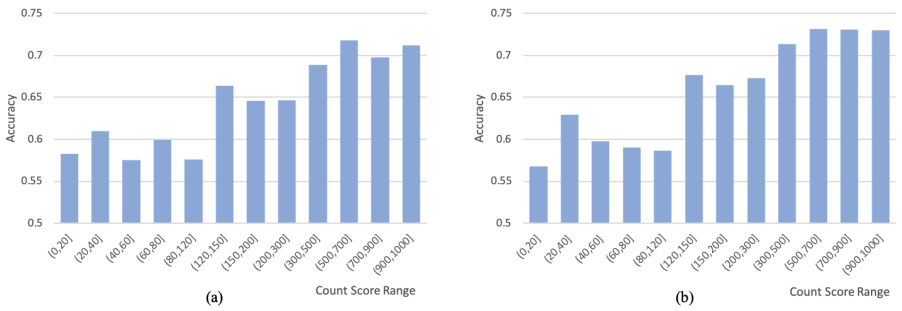

Fig. 4. The average accuracy of proteins within Count score ranges (a) CNN-based prediction model; (b) LSTM-based prediction model.

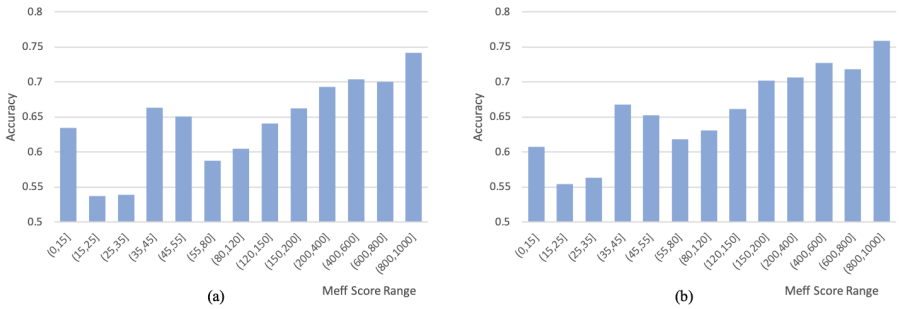

Fig. 5. The average accuracy of proteins within Meff score ranges (a) CNN-based prediction model; (b) LSTM-based prediction model.

Table 1. Number of proteins in certain Count Score ranges.

range	(0,20]	(20,40]	(40,60]	(60,80]	(80,120]	(120,150]	(150,200]	(200,300]	(300,500]	(500,700]	(700,900]	(900,1000]
num	2	16	18	19	29	11	23	27	45	26	26	271

Table 2. Number of proteins in certain Meff Score ranges.

range	(0,15]	(15,25]	(25,35]	(35,45]	(45,55]	(55,80]	(80,120]	(120,150]	(150,200]	(200,400]	(400,600]	(600,800]	(800,1000]
num	12	23	18	9	16	18	19	15	23	68	89	89	114

Enhancement on low-quality PSSM protein Our method is used to enhance the performance of proteins with low-quality PSSM in secondary structure prediction task. However, while improving the low-quality PSSM, noise might have been added to the high-quality PSSM, which would end up with a lower accuracy score. Therefore, we need to find a standard to determine the definition of low-quality proteins for our method, which would be the thresholds of the Count score and the Meff score. As shown in Fig. 6, our method increase or decrease the accuracy of prediction tasks under certain ranges. Greater than 0 means that the average accuracy of our method has improved under the threshold, while less than 0 means that it has decreased. Based on the accuracy results, we are able

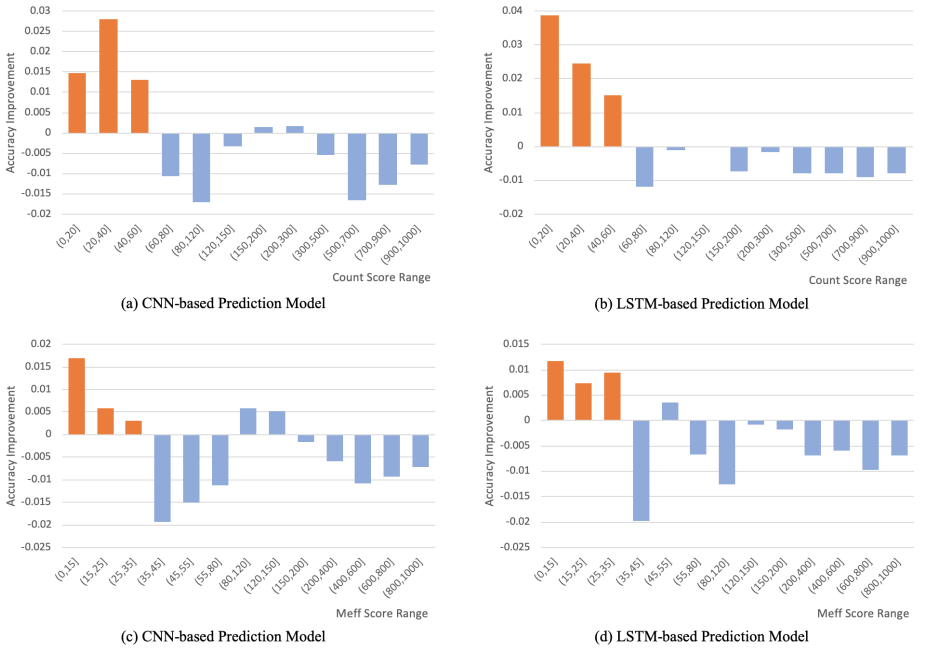

Fig. 6. Our method has achieved significant improvement in all prediction tasks (CNN-based and LSTM-based) when the Count Score is less than 60 (a, b), and the Meff Score is less than 35 (c, d). These figures are the results on CB513 dataset.

to find a consistent trend for both CNN-based and LSTM-based models: our method shows significant superiority for proteins with a Count score less than 60 and a Meff score less than 35.

In addition, in order to verify the threshold we selected is suitable for other datasets, we also report the results of casp11 and casp12, which are shown in table 3. The performances of extensive experiments demonstrate that our method has a significant effect on enhancing low-quality PSSM for different datasets.

Table 3. Comparison results (Q8 accuracy) of our Enhanced PSSM vs. Original PSSM. Enhancement experiments are conducted for low-quality proteins (Count score <= 60 Meff score <= 35) obtained from CB513, CASP11, and CASP12 datasets. Prediction experiments are conducted on CNN-based model and LSTM-based model.

Prediction model	Score range	Datasets	Original PSSM	Enhanced PSSM	Protein num
CNN-based	Count <= 60	CB513	59.106%	61.093%	36
		CASP11	64.196%	67.781%	12
		CASP12	53.300%	56.519%	3
	Meff <= 35	CB513	55.973%	56.717%	53
		CASP11	62.846%	65.732%	17
		CASP12	52.353%	54.462%	7
LSTM-based	Count <= 60	CB513	60.982%	63.041%	36
		CASP11	64.037%	64.990%	12
		CASP12	54.335%	55.865%	3
	Meff <= 35	CB513	56.929%	57.831%	53
		CASP11	63.216%	63.504%	17
		CASP12	51.493%	53.921%	7

5 Conclusion

We propose an innovative Bagging MSA model to enhance low-quality PSSM features of proteins, which would help promote their performance in secondary structure prediction task. We employ an unsupervised learning network to enhance the PSSM features, and two conventional deep learning prediction models as the protein secondary structure prediction networks to prove the effectiveness of our method on various datasets. Our method is the first attempt to enhance PSSM features in the field of protein research. Moreover, the generalization of our Bagging MSA makes it suitable for numerous PSSM related protein prediction tasks. PSSM features are essential for studying proteins, our method pioneer another way to address the prediction limitation for low-quality proteins.

A Appendix

A.1 Encoding networks

As shown in Fig. 7 and Fig. 8, we use 1d-CNN of 3 hidden layers, and 100 neurons for each layer in the local contexts feature encoding module. The window size at each layer is set to 3. And for long-distance module, two stacked bi-LSTM with 512 hidden units are used for all experiments.

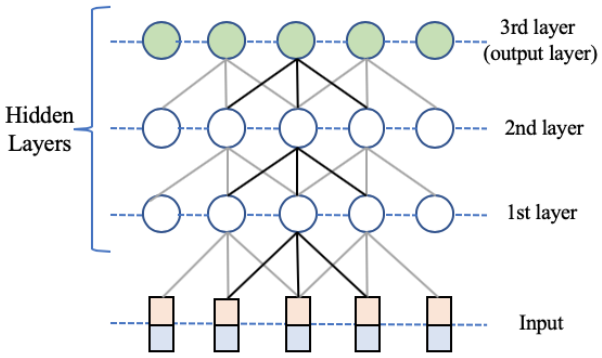

Fig. 7. Local contexts feature encoding module includes three layers of 1d-CNN and the top layer(3rd layer) is the output layer.

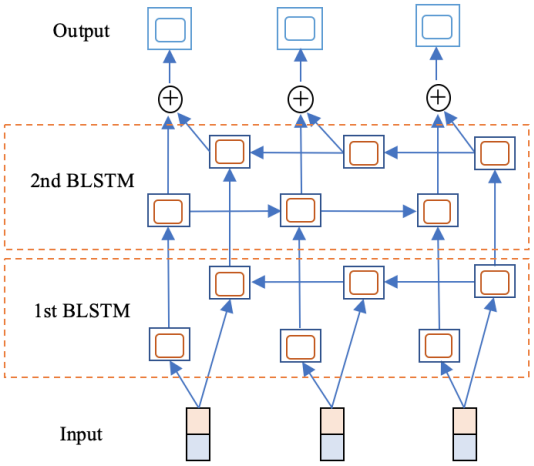

Fig. 8. Long-distance interdependencies feature encoding module includes two stacked BLSTM neural networks.

A.2 Gray-scale images of PSSM

As shown in Fig. 9, which is a set of gray-scale images of the original pssm(a) and enhanced pssm(b) of a protein from cb513 dataset. Where, y-axis is the length N of the protein sequence, the sample protein contains 26 residues(N=26), x-axis is l, 20 plus an unknown amino acids marker(l=21). Lighter colors indicate larger values, while darker colors indicate smaller values. See https://www.rcsb.org for the structure information of the protein(6O4M) in the example.

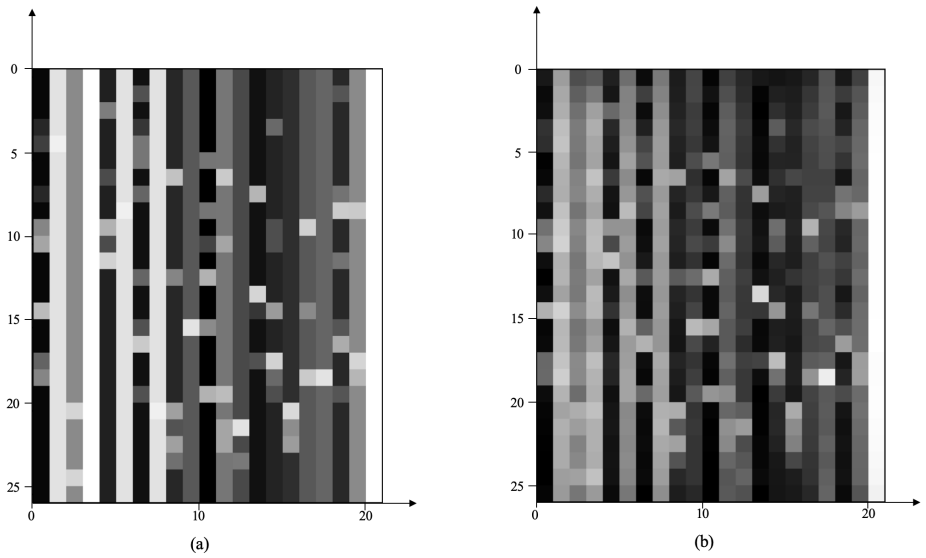

Fig. 9. Gray-scale images of the PSSMs. (a) Original PSSM of 6O4M protein; (b) Enhanced PSSM of 6O4M protein.

A.3 Jackhmmer options for extracting MSA

In the per-target output, report target profiles with an E-value <=1.0; In the per-domain output, for target profiles that have already satisfied the per-profile reporting threshold, report individual domains with a conditional E-value of <=1.0; Use a conditional E-value of <=0.03 as the per-domain inclusion threshold, in targets that have already satisfied the overall per-target inclusion threshold; Obtain residue alignment probabilities from the built-in substitution matrix named BLOSUM62.

A.4 Infrastructure and software

Our model was implemented through Pytorch package. And our models was trained in a self-hosted 16-GPU cluster platform with Intel i7 6700K @ 4.00

GHz CPU, 64 Gigabytes RAM and four Nvidia GTX 1080Ti GPUs on each workstation.

References

1. Heffernan, Rhys, et al. "Improving prediction of secondary structure, local backbone angles, and solvent accessible surface area of proteins by iterative deep learning." Scientific reports 5 (2015): 11476.
2. Morcos, Faruck, et al. "Direct-coupling analysis of residue coevolution captures native contacts across many protein families." Proceedings of the National Academy of Sciences 108.49 (2011): E1293-E1301.
3. Stormo, Gary D., et al. "Use of the 'Perceptron'algorithm to distinguish translational initiation sites in E. coli." Nucleic acids research 10.9 (1982): 2997-3011.
4. Jones, David T. "Protein secondary structure prediction based on position-specific scoring matrices." Journal of molecular biology 292.2 (1999): 195-202.
5. Gao, Yujuan, et al. "RaptorX-Angle: real-value prediction of protein backbone dihedral angles through a hybrid method of clustering and deep learning." BMC bioinformatics 19.4 (2018): 100.
6. Wang, Lusheng, and Tao Jiang. "On the complexity of multiple sequence alignment." Journal of computational biology 1.4 (1994): 337-348.
7. Wang, Zhiyong, and Jinbo Xu. "Predicting protein contact map using evolutionary and physical constraints by integer programming." Bioinformatics 29.13 (2013): i266-i273.
8. Teichert, Florian, et al. "High quality protein sequence alignment by combining structural profile prediction and profile alignment using SABERTOOTH." BMC bioinformatics 11.1 (2010): 251.
9. Remmert, Michael, et al. "HHblits: lightning-fast iterative protein sequence searching by HMM-HMM alignment." Nature methods 9.2 (2012): 173.
10. Wheeler, Travis J., and Sean R. Eddy. "nhmmer: DNA homology search with profile HMMs." Bioinformatics 29.19 (2013): 2487-2489.
11. O'Driscoll, Aisling, et al. "HBLAST: Parallelised sequence similarity–A Hadoop MapReducable basic local alignment search tool." Journal of Biomedical Informatics 54 (2015): 58-64.
12. Wang, Lusheng, and Tao Jiang. "On the complexity of multiple sequence alignment." Journal of computational biology 1.4 (1994): 337-348.
13. Oteri, Francesco, et al. "BIS2Analyzer: a server for co-evolution analysis of conserved protein families." Nucleic acids research 45.W1 (2017): W307-W314.
14. Ju, Fusong, et al. "Seq-SetNet: Exploring Sequence Sets for Inferring Structures." arXiv preprint arXiv:1906.11196 (2019).
15. Ye, Xugang, Guoli Wang, and Stephen F. Altschul. "An assessment of substitution scores for protein profile–profile comparison." Bioinform
16. Altschul, Stephen F., et al. "Gapped BLAST and PSI-BLAST: a new generation of protein database search programs." Nucleic acids research 25.17 (1997): 3389-3402.
17. Morcos, Faruck, et al. "Direct-coupling analysis of residue coevolution captures native contacts across many protein families." Proceedings of the National Academy of Sciences 108.49 (2011): E1293-E1301.
18. Pauling, Linus, Robert B. Corey, and Herman R. Branson. "The structure of proteins: two hydrogen-bonded helical configurations of the polypeptide chain." Proceedings of the National Academy of Sciences 37.4 (1951): 205-211.

19. Kabsch, Wolfgang, and Christian Sander. "Dictionary of protein secondary structure: pattern recognition of hydrogen-bonded and geometrical features." Biopolymers: Original Research on Biomolecules 22.12 (1983): 2577-2637.
20. Suzek, Baris E., et al. "UniRef: comprehensive and non-redundant UniProt reference clusters." Bioinformatics 23.10 (2007): 1282-1288.
21. Mirdita, Milot, et al. "Uniclust databases of clustered and deeply annotated protein sequences and alignments." Nucleic acids research 45.D1 (2016): D170-D176.
22. Allen, David M. "Mean square error of prediction as a criterion for selecting variables." Technometrics 13.3 (1971): 469-475.
23. Gehring, Jonas, et al. "Convolutional sequence to sequence learning." Proceedings of the 34th International Conference on Machine Learning-Volume 70. JMLR. org, 2017.
24. Gehring, Jonas, et al. "A convolutional encoder model for neural machine translation." arXiv preprint arXiv:1611.02344 (2016).
25. Wang, Zhiyong, et al. "Protein 8-class secondary structure prediction using conditional neural fields." 2010 IEEE International Conference on Bioinformatics and Biomedicine (BIBM). IEEE, 2010.
26. Zhou, Jiyun, et al. "CNNH_PSS: protein 8-class secondary structure prediction by convolutional neural network with highway." BMC bioinformatics 19.4 (2018): 60.
27. Dos Santos, Cicero, and Maira Gatti. "Deep convolutional neural networks for sentiment analysis of short texts." Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers. 2014.
28. Wang, Sheng, et al. "Protein secondary structure prediction using deep convolutional neural fields." Scientific reports 6 (2016): 18962.
29. Heffernan, Rhys, et al. "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 (2017): 2842-2849.
30. Hanson, Jack, et al. "Accurate prediction of protein contact maps by coupling residual two-dimensional bidirectional long short-term memory with convolutional neural networks." Bioinformatics 34.23 (2018): 4039-4045.
31. Guo, Yanbu, et al. "DeepACLSTM: deep asymmetric convolutional long short-term memory neural models for protein secondary structure prediction." BMC bioinformatics 20.1 (2019): 341.
32. Zhang, Buzhong, Jinyan Li, and Qiang Lü. "Prediction of 8-state protein secondary structures by a novel deep learning architecture." BMC bioinformatics 19.1 (2018): 293.
33. Andersen, Claus A., Henrik Bohr, and Søren Brunak. "Protein secondary structure: category assignment and predictability." FEBS letters 507.1 (2001): 6-10.
34. Penel, Simon, et al. "Length preferences and periodicity in -strands. Antiparallel edge -sheets are more likely to finish in non-hydrogen bonded rings." Protein engineering 16.12 (2003): 957-961.
35. Sønderby, Søren Kaae, and Ole Winther. "Protein secondary structure prediction with long short term memory networks." arXiv preprint arXiv:1412.7828 (2014).
36. Wang, Guoli, and Roland L. Dunbrack Jr. "PISCES: a protein sequence culling server." Bioinformatics 19.12 (2003): 1589-1591.
37. Zhou, Jian, and Olga G. Troyanskaya. "Deep supervised and convolutional generative stochastic network for protein secondary structure prediction." arXiv preprint arXiv:1403.1347 (2014).
38. Official CASP website, http://predictioncenter.org.

39. Protein Data Bank Homepage, https://www.rcsb.org
40. Bairoch, Amos, et al. "The universal protein resource (UniProt)." Nucleic acids research 33.suppl_1 (2005): D154-D159.
41. Eddy, Sean. "HMMER user's guide." Department of Genetics, Washington University School of Medicine 2.1 (1992): 13.
42. Hochreiter, Sepp, and Jürgen Schmidhuber. "Long short-term memory." Neural computation 9.8 (1997): 1735-1780.
43. Kingma, Diederik P., and Jimmy Ba. "Adam: A method for stochastic optimization." arXiv preprint arXiv:1412.6980 (2014).
44. Touw, Wouter G., et al. "A series of PDB-related databanks for everyday needs." Nucleic acids research 43.D1 (2014): D364-D368.