Received: 5 March 2020 | Revised: 7 July 2020 | Accepted: 2 September 2020

DOI: 10.1002/prot.26007

RESEARCH ARTICLE

PROTEINS WILEY

DNSS2: Improved ab initio protein secondary structure prediction using advanced deep learning architectures

Zhiye Guo[1] | Jie Hou[2] | Jianlin Cheng[1]

[1]Department of Electrical Engineering and Computer Science, University of Missouri, Columbia, Missouri

[2]Department of Computer Science, Saint Louis University, St. Louis, Missouri

Correspondence
Jianlin Cheng, Department of Electrical Engineering and Computer Science, University of Missouri, Columbia, MO 65211.
Email: chengji@missouri.edu

Funding information
National Institute of General Medical Sciences, Grant/Award Number: R01GM093123; National Science Foundation, Grant/Award Numbers: DBI1759934, IIS1763246

Abstract

Accurate prediction of protein secondary structure (alpha-helix, beta-strand and coil) is a crucial step for protein inter-residue contact prediction and ab initio tertiary structure prediction. In a previous study, we developed a deep belief network-based protein secondary structure method (DNSS1) and successfully advanced the prediction accuracy beyond 80%. In this work, we developed multiple advanced deep learning architectures (DNSS2) to further improve secondary structure prediction. The major improvements over the DNSS1 method include (a) designing and integrating six advanced one-dimensional deep convolutional/recurrent/residual/memory/fractal/inception networks to predict 3-state and 8-state secondary structure, and (b) using more sensitive profile features inferred from Hidden Markov model (HMM) and multiple sequence alignment (MSA). Most of the deep learning architectures are novel for protein secondary structure prediction. DNSS2 was systematically benchmarked on independent test data sets with eight state-of-art tools and consistently ranked as one of the best methods. Particularly, DNSS2 was tested on the protein targets of 2018 CASP13 experiment and achieved the Q3 score of 81.62%, SOV score of 72.19%, and Q8 score of 73.28%. DNSS2 is freely available at: https://github.com/multicom-toolbox/DNSS2.

KEYWORDS
CASP, deep learning, secondary structure prediction

1 | INTRODUCTION

Three major types of protein secondary structure are alpha-helix (H), beta-strand (E), and coil state (C),[1] each of which represents the local structure state of an amino acid in a folded polypeptide chain. The predicted information of protein secondary structure is useful for many applications in computational biology, such as protein residue-residue contact prediction,[2-4] protein folding,[5-7] ab-initio protein structure modeling,[8-10] and protein model quality assessment.[11,12] For instance, secondary structure prediction was widely utilized in the template-based structure modeling through threading or comparative modeling on those proteins that have structurally determined homologs,[10,13,14] and in ab initio modeling for those proteins whose sequences share few sequential similarities with known solved structures.[15,16]

The progress in protein secondary structure prediction over the past few decades can be generally summarized from two aspects: the discovery of novel features that are useful for prediction and the development of effective machine learning algorithms.[17,18] The early attempts utilized statistical propensities of single amino acid observed from known structures to identify secondary structures in proteins.[19] The subsequent improvements came from the inclusion of sequence evolutionary profile features inferred from multiple sequence alignment (MSA) such as position-specific scoring matrices (PSSM).[20-25] In addition to the PSSM, the Hidden Markov model (HMM) profiles derived from HHblits[26] was proposed for predicting protein structural

Zhiye Guo and Jie Hou contributed equally to this work

properties.[27] Atchley's factors were also included in some studies to capture the similarity between the types of amino acids.[28,29]

Meanwhile, the machine learning algorithms for protein secondary structure prediction also continued to improve. Several early approaches applied shallow neural networks,[30,31] information theory and Bayesian analysis[32-34] to secondary structure prediction. PSIPRED[21] method proposed a two-stage neural network to predict the secondary structure from the PSI-BLAST sequence profiles. SSpro[24] used bi-directional recurrent neural networks (RNN) to capture the long-range interactions between amino acids. Deep learning techniques recently achieved significant success in secondary structure prediction.[25,29,35-38] DNSS[29] applied an ensemble of deep belief networks to predict 3-state secondary structure. JPRED[39] proposed a novel consensus prediction based upon major voting of the different predictors for secondary structure prediction. SPIDER2[40] employed stacked sparse auto-encoder neural networks to predict the several structural properties iteratively, and this method was further advanced by bidirectional long- and short-term memory (LSTM) neural networks to capture the long-range interactions.[37] DeepCNF[36] integrated the convolutional neural networks (CNN) with conditional random-field to learn the complex sequence-structure relationship and interdependence between sequence and secondary structure. Porter 5.0[41] ensemble seven bidirectional RNN to improve the protein structure prediction. Assisted with the power of deep learning, the accuracy of 3-state secondary structure prediction has been successfully improved above 84%[36-38] on some benchmark data sets. Several studies also made efforts to predict 8-state secondary structure, though more challenging to reach the same accuracy as 3-state secondary structure prediction.[23,36,38,41,42]

In this work, we developed an improved version of our ab initio secondary structure method using multiple advanced deep learning architectures (DNSS2). Three major improvements have been made over the original DNSS method. Firstly, besides the PSSM profile features and Atchley's factors used in DNSS, we incorporated several novel features such as the emission and transition probabilities derived from the HMM profile,[26] and profile probabilities inferred from MSA.[22] All three new features represent the evolutionary conservation information for amino acids in the sequence. Secondly, we designed and integrated six types of advanced one-dimensional deep networks for protein secondary structure prediction, including traditional CNN,[43] recurrent convolutional neural network (RCNN),[44] residual neural network (ResNet),[45] convolutional residual memory networks (CRMN),[46] fractal networks,[47] and Inception network.[48] The ensemble of six networks from DNSS2 significantly improved the secondary structure prediction. Different from the consensus method employed in JPred[39] to acquire the majority voting for the secondary structure type of each amino acid, we simply average the probabilities of different states to make a final prediction of secondary structure class. Besides, we extended the 3-state secondary structure prediction to 8-state prediction that assigns the DSSP 8-class secondary structure to amino acid sequence, including (1) 3-turn 3_{10}-helix (G), (2) 4-turn alpha-helix(H), (3) 5-turn pi-helix (I), (4) hydrogen-bonded turn (T), (5) extended strand in parallel and/or anti-parallel β-sheet conformation (E), (6) β-bridge (B), (7) bend (S) and (8) coil (C). Finally, DNSS2 was trained on a large data set, including 4872 nonredundant protein structures with less than 25% pairwise sequence identity and 2.5 Å resolution. Our method was extensively tested on the independent data set and the latest CASP13 data set with other state-of-art methods and delivered state-of-the-art performance.

2 | MATERIALS AND METHODS

2.1 | Experimental design

In this work, the main objective was to improve the secondary structure prediction by developing more advanced deep learning architectures and introducing more useful features. In the process, we have developed a systematic framework to effectively build deep learning architectures and obtain features to improve secondary structure prediction. Figure 1 provides an overview of our experimental design. Figure 1A lists the six major steps of designing, training, and testing deep learning architectures. Figure 1B illustrates the process of creating training and validation data sets. The key analysis is to design appropriate architectures and investigate if they can improve prediction accuracy. Six different deep neural network architectures were evaluated in the study, including CNN,[43] recurrent RCNN,[44] ResNet,[45] convolutional recurrent memory network (CRMN),[46] FractalNet,[47] and Inception network.[48,49] Most of these architectures were applied to secondary structure prediction for the first time. The detailed description of each network is included in section 2.4. To ensure a fair comparison, each network was optimized using the original feature profiles of training proteins and evaluated on the same validation set of DNSS1. The network that achieved the best Q3 accuracy was selected to explore the feature space on the profiles derived from MSA generated by PSI-BLAST[20] and HHblits,[26] Atchley factors, and emission/transition probabilities inferred from the HMM profile. The optimal feature set was determined according to the highest Q3 accuracy on the validation data sets. The networks were then re-trained using the optimal input profiles to obtain the best models.

Since combining predictors generally improved the prediction accuracy, the different combinations of networks were also evaluated. Finally, after the optimal sets of deep learning architectures and feature profiles were determined, all networks were re-trained on the large data set that was manually curated, including the nonredundant proteins whose structures have been released publicly before 2018. The final networks were used to predict the secondary structure for the test proteins. The probabilities of the three states (ie, helix, sheet, and coil) or eight states (ie, H, G, I, E, B, T, S, C) for each residue predicted by six networks were averaged to make the final secondary structure prediction. Our method was then benchmarked with other state-of-art methods on the independent test data sets.

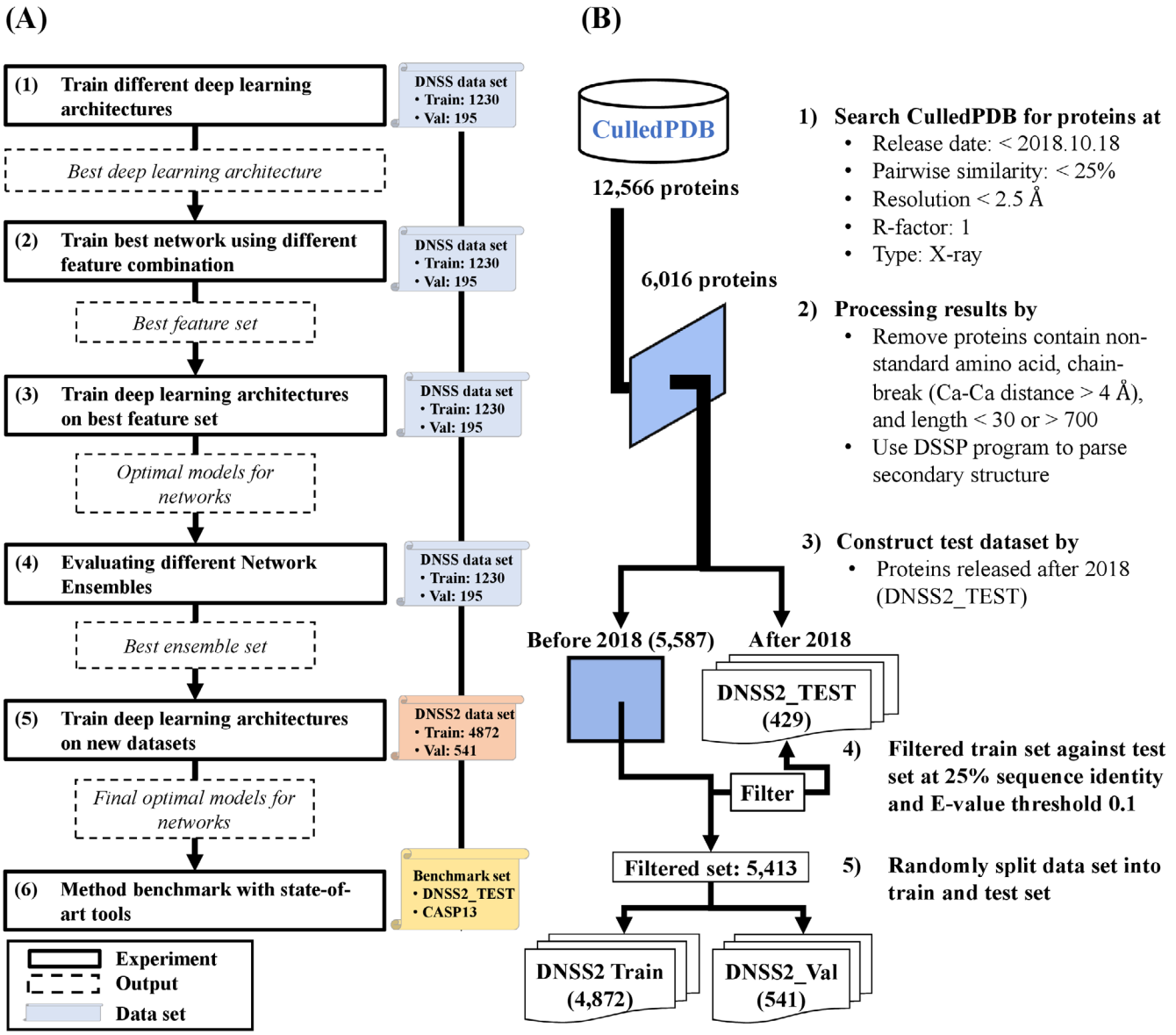

FIGURE 1 Overview of the experimental workflow for improving secondary structure prediction. A, Six principal steps are conducted to construct and train deep networks. The solid box represents an analysis step. The dashed box represents the output from the previous step. The scroll represents the data set used in each step. B, Data set generation and filtering process

2.2 | Data sets and evaluation metric

As described in section 2.1, two training data sets were used in our experiment. In the first stage, the original DNSS data set[29] that included 1230 training proteins and 195 validation proteins was utilized to investigate whether the deep learning architectures and novel features can boost the prediction accuracy.

To utilize more data available since DNSS1 was published, a new, larger training set of DNSS2 was constructed from CullPDB[50] curated on 18 October 2018 (Figure 1B). The data set consists of 12 566 proteins that share less than 25% sequence identity with 2.5 Å resolution cutoff and R-factor cutoff 1. The structures of all the proteins were determined by X-ray crystallography. The data set was then filtered by removing proteins with non-standard amino acids, chain-break (ie, the distance of adjacent Ca-Ca atoms is larger than 4 Å), and sequence length shorter than 30 or longer than 700 amino acids. These threshold parameter values have been widely used in secondary structure prediction methods and other protein bioinformatics works to avoid the sequence redundancy bias and low-quality structures.[2,22] Considering all external methods benchmarked in this work were developed prior to the year 2018, the proteins that were released after 1 January 2018 were extracted as an independent test set (DNSS2_TEST). The resulting set of proteins was further filtered against the DNSS2_TEST set using CD-HIT suite[51] with criteria of 25% sequence identity cutoff and e-value threshold 0.1. Finally, 5413 proteins released prior to 1 January 2018 were obtained as our training set, in which 4872 proteins were used for network training (DNSS2_TRAIN) and 541 proteins were used for model selection

(DNSS2_VAL). In addition, the proteins of the CASP13 (2018) experiment were collected and the ones with at least 25% sequence identity with training proteins were removed, which results in a set of 75 test proteins. The proteins were also classified into template-based (TBM) and free-modeling (FM) targets based on the official CASP definition (CASP 13, 2018, http://www.predictioncenter.org/casp13/index.cgi). We also benchmarked our method on the three publicly available data sets that have been widely used in other studies to evaluate secondary structure prediction, including CB513,[52] CASP11 data set[53] and CASP12 data set.[54] The proteins with at most 25% sequence identity with training proteins were kept for evaluation. It is worth noting that all test data provide true labels of both 3-state and 8-state secondary structure. In summary, the final test sets contain 415 proteins from DNSS2_TEST, 305 proteins from CB513, 71 proteins from CASP11, 36 proteins from CASP12, and 75 proteins from CASP13.

We evaluated our secondary structure prediction based on three primary metrics: Q3, Q8 accuracy, and Segment Overlap measure (SOV). Q3 and Q8 scores represent the percent of correctly predicted secondary structure states in a protein in terms of 3-state and 8-state prediction. SOV score measures the similarity between the predicted segments of continuous structure states and those in the experimental structure.[29,55] The Q3(Q8) and SOV scores are complementary with each other for secondary structure evaluation. All training and testing proteins' structure files were parsed by DSSP program[56] to obtain the real secondary structure classification for each amino acid for training and evaluation. Converting 8-state secondary structure in DSSP into the 3-state secondary structure was implemented in the following ways: (a) treating 3-turn 3_{10}-helix (G) and 5-turn pi-helix (I) as alpha-helix(H), (b) converting bridge (B) into extended strand (E), and (c) replacing the rest types including hydrogen-bonded turn (T)/bend (S) as coil (C).

2.3 | Input features

The profile of each amino acid is represented by 21 numbers from PSI-BLAST-based position specific scoring matrix (PSSM), 20 emission probabilities and 7 transition probabilities extracted from HMM profile, 20 probabilities of standard amino acid calculated from the MSA and 5 numbers derived from Atchley's factor. These features (73 numbers in total) represent the evolutionary conservation and physicochemical properties for residues in a protein sequence.

PSI-BLAST was run to generate MSA and PSSM profile through searching a sequence against filtered UniProt sequence database at 90% sequence identity (UniRef90)[57] with three iterations and an e-value cutoff 0.001 ("-evalue .001 -inclusion_ethresh .002"). Less stringent threshold was used ("-evalue 10 -inclusion_ethresh 10") in case some proteins did not have homologous sequences returned. In a PSSM profile, each position is represented by 20 numbers related to the probabilities for 20 standard amino acids appearing at the position in the MSA. In addition, the sequence information in the second to the last column in PSI-BLAST profile is given for each residue.

HMM profile was generated by running three iteration of "HHblits" against the uniclust30 database (version: October 2017).[58] Two types of probabilities were associated with each residue in an HMM profile: emission probability and transition probability. Emission probability represents the probability of a given amino acid occurring at the position in the MSA. The transition probability represents the probability transiting from an alignment state (ie, match, insertion, and deletion) to another. Similar to PSSM, the emission frequencies of the 20 standard amino acid for each residue were reported in the HMM profile, and the probabilities were calculated according to the formula:

$$p_{ik} = 2^{\left(-\frac{Freq_{ik}}{1000}\right)} \quad (1)$$

where i is the i-th residue in sequence and k is the k-th standard amino acid. And the probability is set to 0 if the frequency is denoted as "*". The transition probabilities for each amino acid were also derived in the same fashion. In total, 20 emission probabilities and 7 transition probabilities for each amino acid were collected to represent the residue conservation inferred from HMM.

Since HHblits was more sensitive to identify distant homologous sequences than PSI-BLAST, the probability matrix of amino acids was also calculated from the MSA generated by HHblits. The conversion from MSA to a probability matrix follows the same calculation as SSpro.[22]

2.4 | Deep learning architectures

A widely used deep learning architecture in bioinformatics is deep CNN. CNN have some distinctive advantages over the traditional neural networks for the bioinformatics problems in several ways: (a) it can learn informative representation directly from sequence features without requiring segmentation (eg, sliding window) or dimension reduction (eg, principal component analysis) techniques; (b) the convolutional network can learn both local and global features to discover complex patterns; and (c) the architecture is independent of input size (ie, length or volume). In this work, we design a standard CNN and five advanced deep learning architectures based on both convolutional and other useful operations as in Figure 2.

Figure 2A illustrates our standard CNN for secondary structure prediction, consisting of a sequence of convolutional blocks, each of which contains a convolutional layer, a batch-normalization layer, and an activation layer. The original input is a L × K vector (X), where L is sequence length and K is the number of features per residue position in the sequence. For each convolution block, the feature maps are obtained after the convolution operation is applied by multiplying the weight matrices (called filters, W) with a window of local features on the previous input layer and adding bias vectors (b) according to the formula: $X^{l+1} = W^{l+1} * X^{l} + b^{l+1}$, where l is the layer number. The batch normalization layer is added to obtain a Gaussian normalization of convolved features coming out of each convolutional layer. Then an activation function such as rectified linear function (ie, ReLU) is applied to extract non-linear patterns of the normalized hidden

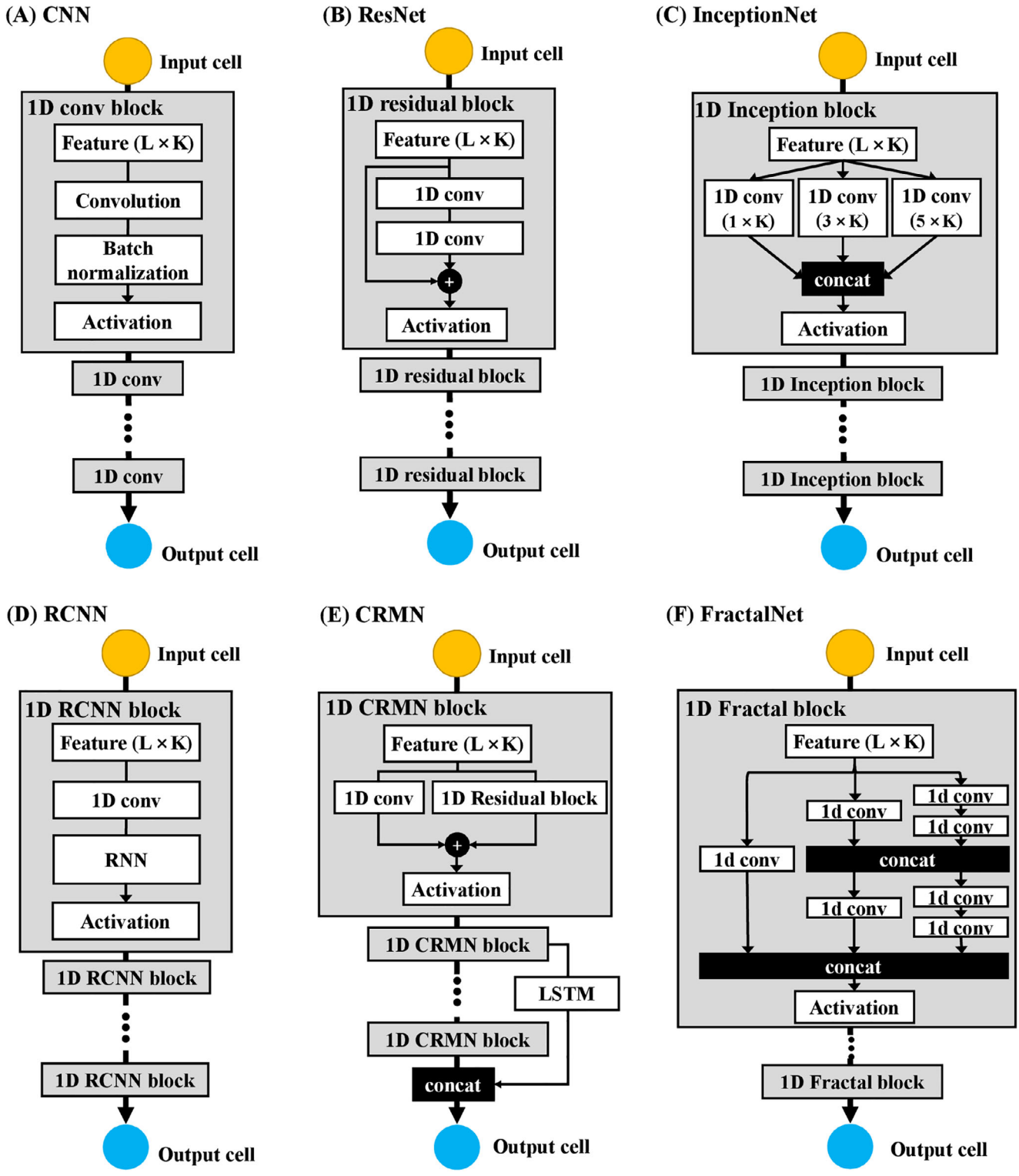

FIGURE 2 Six deep learning architectures: A, CNN, B, ResNet, C, InceptionNet, D, RCNN, E, CRNN, F, FractalNet for secondary structure prediction. L, sequence length; K, number of features per position [Color figure can be viewed at wileyonlinelibrary.com]

features. To avoid overfitting, regularization approaches such as dropout[59] can be applied in the hidden layers. The final output node (also a filter) in the output cell uses the softmax function to classify the input of each residue position from its previous layer into one of three secondary structure states. The output is a L × 3 vector, holding the predicted probability of three secondary structure states for each of L positions in a sequence. The final optimal CNN architecture includes six convolutional blocks, in which the filter size (window size) for each convolutional layer is six, and the number of filters (feature maps) in each convolution layer is 40

The residual network (ResNet) was designed to make traditional CNN deeper without gradient vanishing. The architecture constructs many residual blocks and stacked them up to form a deeper network, as shown in Figure 2B. In each residual block, the input X^l is fed into a few convolutional layers to obtain the nonlinear transformation output $G(X^{l+1})$. In order to make the network deeper, an extra skip connection (ie, short-cut) is added to copy the input X^l to the output of non-linear transformation layer, where $X^{(l+1)^*}$ can be represented as $X^{(l+1)^*} = X^l + G(X^{l+1})$ before applying another ReLU nonlinearity. This process makes neural network deeper by adding shortcuts to facilitate gradient back-propagation during training and achieve better performance. The residual blocks with different configuration can be stacked to achieve higher accuracy. For instance, the final best architecture in DNSS2 is made up of 13 residual blocks, each of which includes three convolutional layers with filter size 1, 3, 1, respectively. The first three residual blocks used 37 filters to learn features, while the middle four blocks used 74 filters for each convolution layer, and the last six residual blocks used 148 filters. In total, 39 convolutional layers are included in the final residual network. In the network, the dropout and batch normalization were also added to prevent network from overfitting.

Inception network is an advanced architecture for building deeper networks by repeating a bunch of inception modules, as shown in Figure 2C. Instead of trying to determine the best values for certain hyper-parameters (ie, number of filter size, number of layers, inclusion of pooling layer), inception network proposes to concatenate outputs of hidden layers with different configuration through an inception module and trains the network to learn patterns from the combination of diverse hyper-parameters. Despite its high computation cost, inception network has performed remarkably well in many applications.[38,48] For secondary structure prediction, a combination of three filter sizes $1 \times K$, $3 \times K$, and $5 \times K$ was applied to convolve feature input, where K is the number of original input features for each residue position. The concatenation of the convolution outputs is fed into an activation layer for non-linear activation calculation. This kind of inception module is repeated to make a deeper network. After the parameter tuning, the optimal inception network comprises three inception blocks with 24 convolution layers included.

In addition, we designed three more deep learning architectures: RCNN,[44] CRMN,[46] and fractal network for secondary structure prediction. The RCNN was designed to model sequential dependency hidden inside the sequential features (Figure 2D), It firstly extracts the higher-level feature maps by a convolution block, and then uses a RNN (ie, bi-directional LSTM network) for modeling the interdependence among the convolved features. Such a recurrent convolutional block with four convolutional layers included is repeated five times to build a deep RCNN for secondary structure prediction in this work. The CRMN network augmented the architectures by integrating convolutional residual networks with LSTM (Figure 2E) (eg, 2 residual blocks and 2 LSTM in the network). Both methods advanced the CNN by introducing the memory mechanisms of RNN. Moreover, inspired by ResNet and Inception Network, we built a Fractal network stacking up different number of convolution blocks in both parallel and hierarchical fashion by adding several shortcut paths to connect lower-level layers and higher-level layers, as shown in Figure 2F. After tuning, the fractal network was assembled with 16 convolution layers for one fractal block.

2.5 | Training and evaluation procedure

Deeper networks with complex architectures are generally difficult to train effectively due to the high-dimensional hyper-parameter space. To obtain good performance on specific feature sets within a reasonable amount of time for each deep network, we developed an efficient heuristic random sampling approach for model hyperparameter optimization. Specifically, based on the several trials on network training, we first determined a reasonable range heuristically for each type of network hyperparameters, including the number of filters from 20 to 50, the number of convolution blocks from 3 to 7, and the filter size from 3 to 7. For each subsequent trial, the values of hyper-parameters were randomly sampled from their specified range, and the Q3 accuracy of the network on the validation data set under the specific parameter combination was assessed. For each deep network, the best parameter set was determined after 100 trials were evaluated. Considering different network architectures have different parameters, on average, each trial took around 6 hours to train on the CPU node with the core "Intel(R) Xeon(R) CPU E5-2680 v4" and ~1 hour on the GPU node with "GeForce GTX 1060 Ti" GPUs each having 6 GB of GPU memory. We found that using the random sampling technique was able to generate better models in most cases and was also more efficient than the traditional grid search or greedy search.

The performance of different deep architectures and different feature profiles on the secondary structure prediction were rigorously examined using the training and validation set from the original DNSS method. After the parameters and input features were determined, we trained each deep network on the latest curated data set (DNSS2_TRAIN) and selected best models using the Q3 accuracy on the independent validation data set (DNSS2_VAL). We used the Keras library (http://keras.io/) along with Tensorflow as a backend to train all networks.

The performance of DNSS2 was evaluated on the independent data sets and compared with a variety of the state-of-art secondary structure prediction tools, including SSpro5.2,[22] PSSpred,[60] MUFOLD-SS,[38] DeepCNF,[36] PSIPRED,[61] SPIDER3,[37] Porter 5,[41] and our previous method DNSS1.[29] All the methods were assessed according to the Q3 (Q8) and SOV scores on each data set.

3 | RESULTS

3.1 | Benchmarking different deep architectures of DNSS2 with DNSS1

The first evaluation was to investigate whether the new deep architectures networks (DNSS2) outperform the deep belief network

(DNSS1) for the secondary structure prediction. In order to fairly compare them, we trained and validated the six deep networks on the original input features of the same 1230 training and 195 validation proteins used to train and test DNSS1. Table 1 compares the Q3 and SOV scores of DNSS1 and DNSS2 architectures on the validation set. The results show that five out of six new advanced deep networks (RCNN, ResNet, CRMN, FractalNet, and InceptionNet) except the standard CNN network obtain higher Q3 scores than the deep belief network that used in DNSS1, while, only FractalNet and InceptionNet achieved higher SOV scores. The different relative performance (ie, higher Q3 score vs lower SOV score) may happen when the predicted secondary structures of residues break the continuous segment in the reference structure. Overall, the InceptionNet worked best among individual deep architectures. The ensemble of the six deep architectures (DNSS2) achieved the highest Q3 score of 83.04% and SOV score of 72.74%, better than or equal to all the six individual deep architectures and 79.1% Q3 score and SOV score of 72.38%, of DNSS1.

3.2 | Impact of different input features

After the best deep learning architecture (ie, InceptionNet) was determined, it was utilized to examine the impact of the different input features including PSSM, Atchley factor (FAC), Emission probabilities (Em), Transition probabilities (Tr), and amino acids probabilities from HHblits alignments (HHblitsMSA). In this analysis, the protein sequence databases required for alignment generation were updated to the latest and all the input features for DNSS1 data sets were regenerated. Specifically, the Uniref90 database that was released in October 2018 was used to generate PSSM profiles by PSI-BLAST, and the latest version of Uniclust30 database (October 2017) was used to generate HMM profiles by HHblits. The Inception network was then trained on the 1230 proteins using the combination of five kinds of features. We tested six feature combinations shown in Table 2. Hyper-parameter optimization was applied to obtain the best model on each feature combination. Table 2 shows the performance of different input feature combinations with the inception network on the validation data set of 195 proteins. Adding the emission profile inferred from HMM model on top of PSSM and Atchley factor features increased the Q3 score from 79.81% to 82.31%. In addition, since PSI-BLAST is a profile-sequence alignment method and HHblits uses both profile-sequence alignment and profile-profile alignment, both methods are able to generate MSA with different sensitivity and specificity. Therefore, combining the features such as PSSM or posterior amino acid substitution probabilities produced by the two complementary methods tend to improve prediction performance. In Table 2, the results show that adding the features derived from the HHblits alignments on top of PSI-BLAST PSSM improves the Q3 accuracy of secondary structure prediction from 79.81% to 81.98% and the SOV score from 71.43% to 74.67%. Finally, Integrating all the five kinds of features will yield the highest Q3 score (ie, 82.72%), with slightly decreased SOV score (75.89%) compared with the best one.

The performance of the six deep architectures and their ensemble on the latest features (the combination of all five kinds of features) of the DNSS1 validation data set was also reported in Table 3. All six architectures were re-trained on the 1230 proteins and evaluated on the validation data set. Compared with the results in Table 1, the prediction accuracy of all the networks on the validation set was improved. The Q3 and SOV scores of the ensemble (DNSS2) were increased to 83.84% and 75.5%, respectively. The results indicate that

TABLE 1 Performance of the six different deep architectures and their ensemble on the DNSS1 validation data set

Method	Q3 (%)	SOV (%)
DNSS1	79.1	72.38
DNSS2_CNN	77.86	68.42
DNSS2_RCNN	79.87	72.34
DNSS2_ResNet	79.61	69.94
DNSS2_CRMN	79.32	69.21
DNSS2_FractalNet	79.85	72.82
DNSS2_InceptionNet	80.68	72.74
DNSS2	83.04	72.74

Note: DNSS2 represents the ensemble of six deep architectures (CNN, RCNN, ResNet, CRMN, FractalNet, and InceptionNet).

TABLE 2 Performance of different input feature combinations on the validation data set of 195 proteins

Rank	Feature name	Q3 (%)	SOV (%)
1	PSSM + FAC + Em + Tr + HHblitsMSA	82.72	75.89
2	PSSM + FAC + Em + Tr	82.36	76.03
3	PSSM + FAC + Em	82.31	74.15
4	PSSM + FAC + HHblitMSA	81.98	74.67
5	PSSM + FAC + Tr	80.13	71.61
6	PSSM + FAC	79.81	71.43

Note: PSSM, FAC, Em, Tr, HHblitsMSA denote five kinds of features: PSSM, atchley factor, emission probabilities, transition probabilities, amino acid probabilities from HHblits alignments.

TABLE 3 Performance of the six different deep learning architectures (CNN, RCNN, ResNet, CRMN, FractalNet, and InceptionNet) and their ensemble (DNSS2) on DNSS1 validation data set and the updated protein sequence database

Method	Q3 (%)	SOV (%)
DNSS2_CNN	80.29	72.1
DNSS2_RCNN	81.83	73.97
DNSS2_ResNet	81.53	73.71
DNSS2_CRMN	81.91	73.37
DNSS2_FractalNet	82.02	73.8
DNSS2_InceptionNet	82.74	75.3
DNSS2	83.84	75.5

TABLE 4 Q3 scores of 9 secondary structure prediction methods on DNSS2_test, CB513, CASP11, and CASP12 data set

	DNSS2_test		CB513		CASP11		CASP12	
Method	**Q3 (%)**	**SOV (%)**	**Q3 (%)**	**SOV (%)**	**Q3 (%)**	**SOV (%)**	**Q3 (%)**	**SOV (%)**
SSPro5.2	79.38	71.02	77.64	69.17	77.72	69.08	76.16	66.59
PSSpred	81.84	71.64	79.60	69.38	79.92	70.57	78.15	67.03
MUFOLD	81.71	73.50	81.05	73.27	81.28	73.91	79.48	69.26
DeepCNF	82.76	70.25	80.87	68.27	81.46	71.40	80.35	67.92
PSIPRED	83.85	74.33	80.40	70.31	82.67	73.88	79.65	70.74
SPIDER3	85.26	77.35	83.48	75.22	83.45	76.09	81.84	71.78
Porter 5	84.92	76.50	83.81	75.25	83.16	75.66	80.58	72.76
DNSS1	80.38	73.89	78.39	71.45	78.59	72.04	76.17	67.40
DNSS2	84.64	75.57	82.56	73.12	82.84	74.34	80.95	71.76

TABLE 5 Comparison of methods on the CASP13 data set in terms of all CASP13 targets, template-based targets, and template-free targets

	All		TBM		FM	
Method	**Q3 (%)**	**SOV (%)**	**Q3 (%)**	**SOV (%)**	**Q3 (%)**	**SOV (%)**
SSPro5.2	76.05	69.13	77.25	69.91	76.12	70.88
PSSpred	78.45	67.07	81.47	71.92	76.99	64.55
MUFOLD	79.88	71.28	81.14	74.53	79.80	70.79
DeepCNF	79.75	68.31	82.21	72.68	78.36	65.55
PSIPRED	80.24	70.87	83.71	75.87	78.41	68.14
SPIDER3	81.20	73.64	84.73	77.93	78.89	71.10
Porter5	81.77	73.85	85.11	78.81	79.42	70.30
DNSS1	76.54	69.29	79.15	72.18	75.46	68.79
DNSS2	81.62	72.19	85.14	76.46	79.82	70.56

the update of the protein sequence databases helps improve prediction accuracy.

3.3 | Comparison of DNSS2 with eight state-of-the-art tools on independent test data sets

DNSS2 was compared with eight state-of-art methods including SSPro5.2, DNSS1, PSSpred, MUFOLD-SS, DeepCNF, PSIPRED, SPIDER3, and Porter 5 on the chosen independent test data sets data set. Particularly, the DNSS_test data set contains nonredundant proteins released after 1 January 2018, which are more likely not included in the training data for all selected state-of-art methods. All the tools were downloaded and configured based on their instructions. The sequence databases that the tools require were updated to the latest version.

The Q3 score of each tool on the test data set was reported in Table 4. In general, DNSS2 is comparable to the two predictors (Porter 5 and SPIDER3) on these data sets and outperforms the other six methods. Specifically, DNSS2 achieved a Q3 accuracy of 84.64% and SOV accuracy of 75.57% on the DNSS2_TEST data set, which was significantly better than DNSS 1.0 on the DNSS2_test data set with *P*-value equal to 2.2E−16.

TABLE 6 Confusion matrix of helix, sheet, and coil predicted by DNSS2 on CASP13 data set

	C pred	E pred	H pred
Coil (C)	80.21%	9.51%	10.28%
Sheet (E)	22.46%	76.45%	1.10%
Helix (H)	11.52%	0.57%	87.91%

Besides, we also compared these methods on the 75 protein targets of the 2018 CASP13 experiment, which share less than 25% sequence identity with the training proteins of DNSS2. Both template-based (TBM) and free-modeling (FM) protein targets were used to evaluate the methods and the results are summarized in Table 5. Consistent with the performance on the test data set shown in Table 4, DNSS2 achieved comparable performance to SPIDER3 and Porter 5. Table 6 summarized the confusion matrix of predictions of three kinds of secondary structures (helix, sheet, coil) by DNSS2 on the CASP13 data set. DNSS2 yields the highest accuracy for helical prediction (87.91%), followed by the coil prediction (80.21%) and the sheet prediction (76.45%). The prediction errors between helix, sheet, and coil were also reported. The error rate of misclassifying helix as sheet is the lowest (0.57%), and sheet as coil is the highest (22.46%).

TABLE 7 Q8 scores of 5-secondary structure prediction methods on DNSS2_test, CB513, CASP11, and CASP12 data set

	DNSS2_TEST		CB513		CASP11		CASP12	
Method	Q8 (%)	SOV (%)	Q8 (%)	SOV (%)	Q8 (%)	SOV (%)	Q8 (%)	SOV (%)
SSPro5.2	69.33	69.71	65.90	66.22	66.87	66.15	64.68	61.80
DeepCNF	70.79	74.25	67.90	70.44	69.15	71.70	67.78	67.53
MUFOLD	70.14	71.03	68.19	70.24	69.05	70.86	66.65	65.38
Porter 5	71.83	74.31	69.41	72.49	70.15	72.65	68.48	69.91
DNSS2	75.46	75.50	73.36	72.91	73.04	73.43	70.82	69.55

TABLE 8 Comparison of 8-state secondary structure prediction methods on the CASP13 data set in terms of all CASP13 targets, template-based targets, and template-free targets

	All		TBM		FM	
Method	Q8 (%)	SOV (%)	Q8 (%)	SOV (%)	Q8 (%)	SOV (%)
SSPro5.2	64.40	64.00	66.36	67.15	63.99	63.71
DeepCNF	66.49	69.23	68.90	72.34	65.32	68.33
MUFOLD	66.70	68.06	68.42	70.94	66.07	68.03
Porter 5	67.71	70.33	70.92	75.37	65.74	67.41
DNSS2	72.72	72.01	75.32	75.99	70.99	70.19

The experimental results demonstrate that DNSS2 can achieve state-of-the-art performance in the 3-state secondary structure prediction. However, we found that DNSS2 outperformed the state-of-the-art methods in terms of 8-state secondary structure prediction. Tables 7 and 8 show the Q8 accuracy of the five selected methods (ie, SSPro, DeepCNF, MUFOLD, Porter5, and DNSS2) that provide the prediction of 8-state secondary structure. According to the results, DNSS2 obtains Q8 scores of 75.46%, 73.36%, 73.04%, 70.82%, and 72.72% on DNSS_test, CB513, CASP11, CASP12, and CASP13 data set respectively, consistently outperforming all the other four methods and achieving ~2% to ~4% improvements. Particularly, for those CASP13 free-modeling targets that do not have protein homologs, DNSS2 can also achieve 70.99% Q8 accuracy and 70.19% SOV score, while the highest scores delivered by other methods in this subset are 66.07% and 68.33% for Q8 and SOV score, respectively. The analysis demonstrates that DNSS2 delivers the best performance for 8-state secondary structure prediction.

4 | CONCLUSION

In this work, we developed several advanced deep learning architectures and their ensemble to improve secondary structure prediction. We investigated six advanced deep learning architectures and five kinds of input features on secondary structure prediction. Several deep learning architectures such as fractal network, and convolutional residual memory network are novel for protein secondary structure prediction, and performed better than the deep belief network. The performance of the deep learning method is comparable to or better than seven external state-of-the-art methods on the several independent test data sets, especially for the 8-state secondary structure prediction. Our experiment also demonstrated that emission/transition probabilities extracted from HMM profiles are useful for secondary structure prediction.

ACKNOWLEDGMENTS

This work has been supported by an NIH grant (R01GM093123) and two NSF grants (DBI1759934, IIS1763246) to J. C.

CONFLICT OF INTERESTS

The authors have no conflict of interests to declare.

PEER REVIEW

The peer review history for this article is available at https://publons.com/publon/10.1002/prot.26007.

ORCID

Jianlin Cheng https://orcid.org/0000-0003-0305-2853

REFERENCES

1. Pauling L, Corey RB, Branson HR. The structure of proteins: two hydrogen-bonded helical configurations of the polypeptide chain. *Proc Natl Acad Sci*. 1951;37(4):205-211.
2. Wang S, Sun S, Li Z, Zhang R, Xu J. Accurate de novo prediction of protein contact map by ultra-deep learning model. *PLoS Comput Biol*. 2017;13(1):e1005324.
3. Adhikari B, Hou J, Cheng J. DNCON2: improved protein contact prediction using two-level deep convolutional neural networks. *Bioinformatics*. 2017;34(9):1466-1472.
4. Michel M, Hurtado DM, Elofsson A. PconsC4: fast, accurate, and hassle-free contact predictions. *Bioinformatics*. 2018;35(15):2677-2679.
5. Hou J, Adhikari B, Cheng J. DeepSF: deep convolutional neural network for mapping protein sequences to folds. *Bioinformatics*. 2018;34(8):1295-1303.

6. Jones DT, Tress M, Bryson K, Hadley C. Successful recognition of protein folds using threading methods biased by sequence similarity and predicted secondary structure. *Proteins*. 1999;37(S3):104-111.
7. Myers JK, Oas TG. Preorganized secondary structure as an important determinant of fast protein folding. *Nat Struct Mol Biol*. 2001;8(6):552-558.
8. Adhikari B, Cheng J. CONFOLD2: improved contact-driven ab initio protein structure modeling. *BMC Bioinformatics*. 2018;19(1):22.
9. Rohl CA, Strauss CE, Misura KM, Baker D. Protein structure prediction using Rosetta. *Methods Enzymol*. 2004;383:66-93.
10. Roy A, Kucukural A, Zhang Y. I-TASSER: a unified platform for automated protein structure and function prediction. *Nat Protoc*. 2010;5(4):725-738.
11. Uziela K, Shu N, Wallner B, Elofsson A. ProQ3: improved model quality assessments using Rosetta energy terms. *Sci Rep*. 2016;6:33509.
12. Cao R, Cheng J. Integrated protein function prediction by mining function associations, sequences, and protein–protein and gene–gene interaction networks. *Methods*. 2016;93:84-91.
13. Webb B, Sali A. Protein structure modeling with MODELLER. *Prot Struct Predict*. 2014;1137:1-15.
14. Wang Z, Eickholt J, Cheng J. MULTICOM: a multi-level combination approach to protein structure prediction and its assessments in CASP8. *Bioinformatics*. 2010;26(7):882-888.
15. Kryshtafovych A, Monastyrskyy B, Fidelis K, Moult J, Schwede T, Tramontano A. Evaluation of the template-based modeling in CASP12. *Proteins*. 2018;86(S1):321–334.
16. Ovchinnikov S, Park H, Kim DE, DiMaio F, Baker D. Protein structure prediction using Rosetta in CASP12. *Proteins*. 2018;86(S1):113–121.
17. Yang Y, Gao J, Wang J, et al. Sixty-five years of the long march in protein secondary structure prediction: the final stretch? *Brief Bioinform*. 2018;19(3):482–494.
18. Rost B. Protein secondary structure prediction continues to rise. *J Struct Biol*. 2001;134(2–3):204-218.
19. Chou PY, Fasman GD. Prediction of protein conformation. *Biochemistry*. 1974;13(2):222-245.
20. Altschul SF, Madden TL, Schäffer AA, et al. PSI-BLAST: a new generation of protein database search programs. *Nucleic Acids Res*. 1997;25(17):3389-3402.
21. Jones DT. Protein secondary structure prediction based on position-specific scoring matrices. *J Mol Biol*. 1999;292(2):195-202.
22. Magnan CN, Baldi P. SSpro/ACCpro 5: almost perfect prediction of protein secondary structure and relative solvent accessibility using profiles, machine learning and structural similarity. *Bioinformatics*. 2014;30(18):2592-2597.
23. Pollastri G, Mclysaght A. Porter: a new, accurate server for protein secondary structure prediction. *Bioinformatics*. 2004;21(8):1719-1720.
24. Pollastri G, Przybylski D, Rost B, Baldi P. Improving the prediction of protein secondary structure in three and eight classes using recurrent neural networks and profiles. *Proteins*. 2002;47(2):228-235.
25. Dor O, Zhou Y. Achieving 80% ten-fold cross-validated accuracy for secondary structure prediction by large-scale training. *Proteins*. 2007;66(4):838-845.
26. Remmert M, Biegert A, Hauser A, Söding J. HHblits: lightning-fast iterative protein sequence searching by HMM-HMM alignment. *Nat Methods*. 2012;9(2):173-175.
27. Meng Q, Peng Z, Yang J, Valencia A. CoABind: a novel algorithm for coenzyme a (CoA)-and CoA derivatives-binding residues prediction. *Bioinformatics*. 2018;1:7.
28. Atchley WR, Zhao J, Fernandes AD, Drüke T. Solving the protein sequence metric problem. *Proc Natl Acad Sci*. 2005;102(18):6395-6400.
29. Spencer M, Eickholt J, Cheng J. A deep learning network approach to ab initio protein secondary structure prediction. *IEEE/ACM Trans Comput Biol Bioinform*. 2015;12(1):103-112.
30. Qian N, Sejnowski TJ. Predicting the secondary structure of globular proteins using neural network models. *J Mol Biol*. 1988;202(4):865-884.
31. Holley LH, Karplus M. Protein secondary structure prediction with a neural network. *Proc Natl Acad Sci*. 1989;86(1):152-156.
32. Gibrat J-F, Garnier J, Robson B. Further developments of protein secondary structure prediction using information theory: new parameters and consideration of residue pairs. *J Mol Biol*. 1987;198(3):425-443.
33. Stolorz P, Lapedes A, Xia Y. Predicting protein secondary structure using neural net and statistical methods. *J Mol Biol*. 1992;225(2):363-377.
34. Schmidler SC, Liu JS, Brutlag DL. Bayesian segmentation of protein secondary structure. *J Comput Biol*. 2000;7(1–2):233-248.
35. Faraggi E, Zhang T, Yang Y, Kurgan L, Zhou Y. SPINE X: improving protein secondary structure prediction by multistep learning coupled with prediction of solvent accessible surface area and backbone torsion angles. *J Comput Chem*. 2012;33(3):259-267.
36. Wang S, Peng J, Ma J, Xu J. Protein secondary structure prediction using deep convolutional neural fields. *Sci Rep*. 2016;6(1):1-11.
37. 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(18):2842-2849.
38. Fang C, Shang Y, Xu D. MUFOLD-SS: new deep inception-inside-inception networks for protein secondary structure prediction. *Proteins*. 2018;86(5):592-598.
39. Drozdetskiy A, Cole C, Procter J, Barton GJ. JPred4: a protein secondary structure prediction server. *Nucleic Acids Res*. 2015;43(W1):W389-W394.
40. Heffernan R, Paliwal K, Lyons J, et al. Improving prediction of secondary structure, local backbone angles, and solvent accessible surface area of proteins by iterative deep learning. *Sci Rep*. 2015;5:11476.
41. Torrisi M, Kaleel M, Pollastri G. Porter 5: fast, state-of-the-art ab initio prediction of protein secondary structure in 3 and 8 classes. *bioRxiv*. 2018;289033. https://www.biorxiv.org/content/10.1101/289033v4.full.
42. Zhang B, Li J, Lü Q. Prediction of 8-state protein secondary structures by a novel deep learning architecture. *BMC Bioinformatics*. 2018;19(1):293.
43. Krizhevsky A, Sutskever I, Hinton GE. *Imagenet Classification with Deep Convolutional Neural Networks*. Harrahs and Harveys, Lake Tahoe: Advances in neural information processing systems; 2012:1097-1105.
44. Liang M, Hu, X. *Recurrent convolutional neural network for object recognition*. Paper presented at Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2015; pp. 3367–3375.
45. He K, Zhang X, Ren S, Sun J. Deep residual learning for image recognition. Paper presented at: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016; pp. 770–778.
46. Moniz J, Pal C. Convolutional residual memory networks. *arXiv preprint arXiv*. 2016;1606.05262. https://arxiv.org/abs/1606.05262.
47. Larsson G, Maire M, Shakhnarovich G. Fractalnet: Ultra-deep neural networks without residuals. *arXiv preprint arXiv*. 2016;1605:07648. https://arxiv.org/abs/1605.07648.
48. Szegedy C, Liu W, Jia Y, et al. Going deeper with convolutions. Paper presented at: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2015; pp. 1–9.
49. Fang C, Li Z, Xu D, Shang Y. MUFold-SSW: a new web server for predicting protein secondary structures, torsion angles and turns. *Bioinformatics*. 2020;36(4):1293-1295.
50. Wang G, Dunbrack RL Jr. PISCES: a protein sequence culling server. *Bioinformatics*. 2003;19(12):1589-1591.

51. Li W, Godzik A. Cd-hit: a fast program for clustering and comparing large sets of protein or nucleotide sequences. *Bioinformatics*. 2006;22 (13):1658-1659.
52. Zhou, J, Troyanskaya, O. *Deep supervised and convolutional generative stochastic network for protein secondary structure prediction*. Paper presented at: International conference on machine learning, 2014; pp. 745–753.
53. Moult J, Fidelis K, Kryshtafovych A, Schwede T, Tramontano A. Critical assessment of methods of protein structure prediction: Progress and new directions in round XI. *Proteins*. 2016;84:4-14.
54. Moult J, Fidelis K, Kryshtafovych A, Schwede T, Tramontano A. Critical assessment of methods of protein structure prediction (CASP)—round XII. *Proteins*. 2018;86:7-15.
55. Zemla A, Venclovas Č, Fidelis K, Rost B. A modified definition of Sov, a segment-based measure for protein secondary structure prediction assessment. *Proteins*. 1999;34(2):220-223.
56. Kabsch W, Sander C. Dictionary of protein secondary structure: pattern recognition of hydrogen-bonded and geometrical features. *Biopolymers*. 1983;22(12):2577-2637.
57. Consortium U. UniProt: a hub for protein information. *Nucleic Acids Res*. 2014;43(D1):D204-D212.
58. Mirdita M, von den Driesch L, Galiez C, Martin MJ, Söding J, Steinegger M. Uniclust databases of clustered and deeply annotated protein sequences and alignments. *Nucleic Acids Res*. 2016;45(D1): D170-D176.
59. Srivastava N, Hinton GE, Krizhevsky A, Sutskever I, Salakhutdinov R. Dropout: a simple way to prevent neural networks from overfitting. *J Mach Learn Res*. 2014;15(1):1929-1958.
60. Yan R, Xu D, Yang J, Walker S, Zhang Y. A comparative assessment and analysis of 20 representative sequence alignment methods for protein structure prediction. *Sci Rep*. 2013;3:2619.
61. McGuffin LJ, Bryson K, Jones DT. The PSIPRED protein structure prediction server. *Bioinformatics*. 2000;16(4):404-405.

How to cite this article: Guo Z, Hou J, Cheng J. DNSS2: Improved ab initio protein secondary structure prediction using advanced deep learning architectures. *Proteins*. 2021;89: 207–217. https://doi.org/10.1002/prot.26007