Bioinformatics, 34(8), 2018, 1295–1303
doi: 10.1093/bioinformatics/btx780
Advance Access Publication Date: 8 December 2017
Original Paper

Structural bioinformatics

DeepSF: deep convolutional neural network for mapping protein sequences to folds

Jie Hou[1], Badri Adhikari[2] and Jianlin Cheng[1,3,*]

[1]Department of Electrical Engineering and Computer Science, University of Missouri, Columbia, MO 65211, USA, [2]Department of Mathematics and Computer Science, University of Missouri-St. Louis, St. Louis, MO 63121, USA and [3]Informatics Institute, University of Missouri, Columbia, MO 65211, USA

*To whom correspondence should be addressed.

Associate Editor: Alfonso Valencia

Received on April 1, 2017; revised on November 10, 2017; editorial decision on November 29, 2017; accepted on December 7, 2017

Abstract

Motivation: Protein fold recognition is an important problem in structural bioinformatics. Almost all traditional fold recognition methods use sequence (homology) comparison to indirectly predict the fold of a target protein based on the fold of a template protein with known structure, which cannot explain the relationship between sequence and fold. Only a few methods had been developed to classify protein sequences into a small number of folds due to methodological limitations, which are not generally useful in practice.

Results: We develop a deep 1D-convolution neural network (DeepSF) to directly classify any protein sequence into one of 1195 known folds, which is useful for both fold recognition and the study of sequence–structure relationship. Different from traditional sequence alignment (comparison) based methods, our method automatically extracts fold-related features from a protein sequence of any length and maps it to the fold space. We train and test our method on the datasets curated from SCOP1.75, yielding an average classification accuracy of 75.3%. On the independent testing dataset curated from SCOP2.06, the classification accuracy is 73.0%. We compare our method with a top profile–profile alignment method—HHSearch on hard template-based and template-free modeling targets of CASP9-12 in terms of fold recognition accuracy. The accuracy of our method is 12.63–26.32% higher than HHSearch on template-free modeling targets and 3.39–17.09% higher on hard template-based modeling targets for top 1, 5 and 10 predicted folds. The hidden features extracted from sequence by our method is robust against sequence mutation, insertion, deletion and truncation, and can be used for other protein pattern recognition problems such as protein clustering, comparison and ranking.

Availability and implementation: The DeepSF server is publicly available at: http://iris.rnet.missouri.edu/DeepSF/.

Contact: chengji@missouri.edu

Supplementary information: Supplementary data are available at *Bioinformatics* online.

1 Introduction

Protein folding reveals the evolutionary process between the protein amino acid sequence and its atomic tertiary structure (Dill *et al.*, 2008). Folds represent the main characteristics of protein structures, which describe the unique arrangement of secondary structure elements in the infinite conformation space (Hadley and Jones, 1999; Murzin *et al.*, 1995). Several fold classification databases such as SCOP (Murzin *et al.*, 1995), CATH (Greene *et al.*, 2007), FSSP (Holm and Sander, 1994), ECOD (Cheng *et al.*, 2014) have been developed to summarize the structural relationship between

proteins. With the substantial investment in protein structure determination in the past decades, the number of experimentally determined protein structures has substantially increased to more than 100 000 in the Protein Data Bank (PDB) (Berman *et al.*, 2000; Murzin *et al.*, 1995). However, due to the conservation of protein structures, the number of unique folds has been rather stable. For example, the SCOP 1.75 curated in 2009 has 1195 unique folds, whereas SCOP 2.06 only has 26 more folds identified from the recent PDB (Chandonia *et al.*, 2016). Generally, determining the folds of a protein can be accomplished by comparing its structure with those of other proteins whose folds are known. However, because the structures of most (>99%) proteins are not known, the development of sequence-based computational fold detection method is necessary and essential to automatically assign proteins into fold. And identifying protein homologs sharing the same fold is a crucial step for computational protein structure predictions (Jo *et al.*, 2015; Söding, 2005) and protein function prediction (Cao and Cheng, 2016).

Sequence-based methods for protein fold recognition can be summarized into two categories: (i) sequence alignment methods and (ii) machine learning method. The sequence alignment methods (Altschul *et al.*, 1990; Henikoff and Henikoff, 1992) align the sequence of a target protein against the sequences of template proteins whose folds are known to generate alignment scores. If the score between a target and a template is significantly higher than that of two random sequences, the fold of the template is considered to be the fold of the target. In order to improve the sensitivity of detecting remote homologous sequences that share the same fold, sequence alignment methods were extended to align the profiles of two proteins. Profile–sequence alignment method (Altschul *et al.*, 1997) and profile–profile alignment methods based hidden Markov model (HMM) (Söding, 2005) or Markov random fields (MRFs) (Ma *et al.*, 2014) are more sensitive in recognize proteins that have the same fold, but little sequence similarity, than sequence–sequence alignment methods. Despite the success, the sequence alignment methods are essentially an indirect fold recognition approach that transfers the fold of the nearest sequence neighbors to a target protein, which cannot explain the sequence–structure relationship of the protein.

Machine learning methods have been developed to directly classify proteins into different fold categories (Chung *et al.*, 2003; Damoulas and Girolami, 2008; Dong *et al.*, 2009; Wei *et al.*, 2015). Multi-layer perception and support vector machine have been used to construct a single classifier to recognize fold pattern in an early work (Chung *et al.*, 2003). Ensemble classifiers were proposed to improve fold recognition (Shen and Chou, 2006). In order to better use sequence features, kernel-based learning was designed to classify protein folds (Damoulas and Girolami, 2008). A recent ensemble-based method combined template-based search and support vector machine classification to recognize protein folds (Xia *et al.*, 2016). However, because traditional machine learning methods cannot classify data into a large number of categories (e.g. thousands of folds), these methods can only classify proteins into a small number (e.g. dozens) of pre-selected fold categories, which cannot be generally applied to predict the fold of an arbitrary protein and therefore is not practically useful for protein structure prediction. To work around the problem, another kind of machine learning methods (Cheng and Baldi, 2006; Jo and Cheng, 2014; Jo *et al.*, 2015) converts a multi-fold classification problem into a binary classification problem to predict if a target protein and a template protein share the same fold based on their pairwise similarity features, which is still an indirect approach that cannot directly explain how a protein sequence is mapped to one of thousands of folds in the fold space.

In this work, we utilize the enormous learning power of deep learning to directly classify any protein into one of 1195 known folds. Deep learning techniques have achieved significant success in computer vision, speech recognition and natural language processing (Kim, 2014; Krizhevsky *et al.*, 2012). The application of deep learning in bioinformatics has also gained the traction since 2012. Deep belief networks (Eickholt and Cheng, 2012) were developed to predict protein residue-residue contacts. Recently a deep residual convolutional neural network was designed to further improve the accuracy of contact prediction (Wang *et al.*, 2017). Deep learning methods have also been applied to predict protein secondary structures (Spencer *et al.*, 2015; Wang *et al.*, 2016) and identify protein pairs that have the same fold (Jo *et al.*, 2015; Ma *et al.*, 2014).

Here, we design a one-dimensional (1D) deep convolution neural network method (DeepSF) to classify proteins of variable-length into all 1195 known folds defined in SCOP 1.75 database. DeepSF can directly extract hidden features from any protein sequence of any length through convolution transformation, and then classify it into one of thousands of folds accurately. The method is the first method that can map all protein sequences in the sequence space directly into all the folds in the fold space without relying on pairwise sequence comparison (alignment). The hidden fold-related features generated from sequences can be used to measure the similarity between proteins, cluster proteins and select template proteins for tertiary structure prediction.

We rigorously evaluated our method on three test datasets: new proteins in SCOP 2.06 database, template-based targets in the past CASP experiments, and template-free targets in the past CASP experiments. Our method (DeepSF) is more sensitive than a state-of-the-art profile–profile alignment method—HHSearch in predicting the fold of a protein, and it is also much faster than HHSearch because it directly classifies a protein into folds without searching a template database (see Section 8 in the Supplementary Material). We also demonstrate that the hidden features extracted from protein sequences by DeepSF is robust against residue mutation, insertion, deletion and truncation. To generalize the application of our method, we also applied our deep convolutional neural network to classify proteins based on ECOD domain classification database (Cheng *et al.*, 2014), which focuses on distant evolutionary relationships between proteins.

2 Materials and methods

2.1 Datasets

2.1.1 Training, validation and test datasets

The main dataset that we used for training, validation and test was downloaded from the SCOP 1.75 genetic domain sequence subsets with less than 95% pairwise identity released in 2009. The protein sequences for each SCOP domain were cleaned according to the observed residues in the atomic structures (Murzin *et al.*, 1995). The dataset contains 16 712 proteins covering 7 major structural classes with total 1195 identified folds. The number of proteins in each fold is very uneven, with 5% (i.e. 61/1195) folds each having >50 proteins, 26% (i.e. 314/1195) folds each having 6 to 50 proteins, and 69% (820/1195) each having ≤5 proteins (Supplementary Fig. S1), making it challenging to train a classifier accurately predicting all the folds, especially small folds with few protein sequences. The proteins in all 1195 folds have sequence length ranging from 9 to 1419 (Supplementary Fig. S2a), and most of them have length in the range

of 9–600 (Supplementary Fig. S2b). In order to remove the homologous sequence redundancy between test datasets and training datasets, we adopted two different strategies for homology reduction: three-level redundancy removal at fold/superfamily/family levels and sequence identity reduction. The three-level redundancy removal started with fold-level reduction that split proteins into a fold-level training dataset and a fold-level test dataset based on superfamilies, i.e. no proteins from the same superfamily will be included in both training and test datasets. The fold-level training dataset was split into a superfamily-level training dataset and a superfamily-level test dataset based on families, i.e. no proteins from the same family existed in both the training and test datasets. Finally, the superfamily-level training dataset was split into a family-level training dataset and a family level test dataset by sampling 80% of proteins in the same family for training and using the remaining 20% for test. After the three-level reduction, the 80% of proteins sampled from the fold-level, superfamily-level and family-level test datasets, respectively, were combined into one test dataset. The remaining 20% of proteins from the fold-level, superfamily-level and family-level test datasets were combined into a validation dataset. We further removed the proteins in the validation dataset whose E-value of sequence similarity with proteins in the training dataset is less than '1e-4'. More detailed description about three-level homology removal and how to tune hyper parameters on the validation dataset can found in Section 1.1 in the Supplementary Material. The distribution of E-value of best hits for proteins in the validation and test datasets in terms of family, superfamily and fold level is shown in Figure S7 in the Supplementary Material. The three-level test datasets can validate the performance of the method at fold, superfamily and family level on SCOP 1.75 database, respectively.

In order to validate the performance on two independent datasets: SCOP 2.06 (see Section 2.1.2) and CASP dataset (see Section 2.1.3), the SCOP 1.75 dataset with less than or equal to 95% sequence identity was split into a training dataset and a validation set according 8/2 ratio for each fold. The validation dataset was further filtered to at most 70, 40, 25% pairwise similarity with the training dataset according to the sequence identity reduction (see details for sequence similarity reduction in Section 1.2 in the Supplementary Material).

2.1.2 Independent SCOP 2.06 test dataset

In order to independently test the performance of our method, we collected the protein sequences in the latest SCOP 2.06 (Chandonia *et al.*, 2016), but not in SCOP 1.75. The sequences with similarity greater than 40% with SCOP 1.75 dataset were further removed. And the remaining proteins were filtered to less than or equal to 25% pairwise similarity with e-value cutoff '1e-4' by CD-Hit suite (Li and Godzik, 2006). The parameter setting for CD-HIT is described in Section 9.1. in the Supplementary Material. Finally, this independent SCOP test dataset contains 2533 domains, covering 550 folds, which were split into three sub test datasets (37 proteins in the fold-level test dataset, 1754 in the super-family level test dataset and 742 in the family-level test dataset).

2.1.3 Independent CASP test dataset

Besides classifying the proteins with known folds in the SCOP, we tested our methods on a protein dataset consisting of template-free and template-based targets used in the 9th, 10th, 11th and 12th Critical Assessments of Structure Prediction (CASP) experiments from 2010 to 2016 (Kinch *et al.*, 2016; Kinch *et al.*, 2011). These are new proteins available after SCOP 1.75 was created in 2009. The complete CASP dataset contains 431 domains. The sequences in the CASP dataset with sequence identity >10% against the SCOP training dataset are removed. To assign the folds to these CASP targets, we compare each CASP target against all domains in SCOP 1.75 using the structural similarity metric-TM-score (Zhang and Skolnick, 2005). Based on the evaluation of domains from each fold, referred to Supplementary Section 2, if a CASP target has TM-score above 0.5 with a SCOP domain, with 0.67 percentage alignment and RMSD < 3.57, suggesting they have the same fold, the fold of the SCOP domain is transferred to the CASP target (Xu and Zhang, 2010). If the CASP target does not have the same fold with any SCOP domain, it is removed from the dataset. After preprocessing, the dataset has 183 protein targets with fold assignment, which include 95 template-free (FM) or seemly template-free (FM/TBM) targets and 88 template-based (TBM) targets, where the categories of targets were defined by CASP experiments (Kinch *et al.*, 2011).

2.2 Input feature generation and label assignment

We generated four kinds of input features representing the (i) sequence, (ii) profile, (iii) predicted secondary structure and (iv) predicted solvent accessibility of each protein. Each residue in a sequence is represented as a 20-dimension zero-one vector in which only the value at the residue index is marked as 1 and all others are marked as 0. The position-specific scoring matrix (PSSM) for each sequence is calculated by using PSI-BLAST to search the sequence against the 'nr90' database. The 20 numbers in the PSSM corresponding to each position in the protein sequence is used as features to represent the profile of amino acids at the position. We predicted 3-class secondary structure (Helix, Strand, Loop) and two-class solvent accessibility (Exposed, Buried) for each protein sequence using SCRATCH (Magnan and Baldi, 2014). The secondary structure of each position is represented by 3 binary numbers with one of them as 1, indicating which secondary structure it is. Similarly, the solvent accessibility at each position is denoted by two binary numbers. In total, each position of a protein sequence is represented by a vector of 45 numbers. The whole protein is encoded by $L \times 45$ numbers. It is worth noting that these input features have been used in protein fold recognition. (Damoulas and Girolami, 2008; Jo et al., 2015; Xia et al., 2016). Each sequence is assigned to a pre-defined fold index in the range of 0 ~1194 denoting its fold according to SCOP 1.75 definition, which is the class label of the protein.

2.3 Deep convolutional neural network for fold classification

The architecture of the deep convolutional neural network for mapping protein sequences to folds (DeepSF) is shown in Figure 1. It contains 15 layers including input layer, 10 convolutional layers, one K-max pooling layer, one flattening layer, one fully-connected hidden layer and an output layer. The softmax function is applied to the nodes in the output layer to predict the probability of 1195 folds. The input layer has $L \times 45$ input numbers representing the positional information of a protein sequence of variable length L. Each of 10 filters in the first convolution layer is applied to the windows in the input layer to generate $L \times 1$ hidden features (feature map) through the convolution operation, batch-normalization and non-linear transformation of its inputs with the rectified-linear unit (ReLU) activation function (Krizhevsky *et al.*, 2012), resulting $10 \times L$ hidden features. Different window sizes (i.e. filter size) in the 1D convolution layer are tested and finally two window sizes (6 and 10) are chosen, which are close to the average length of beta-sheet and alpha-helix in a protein. The hidden features generated by 10 filters

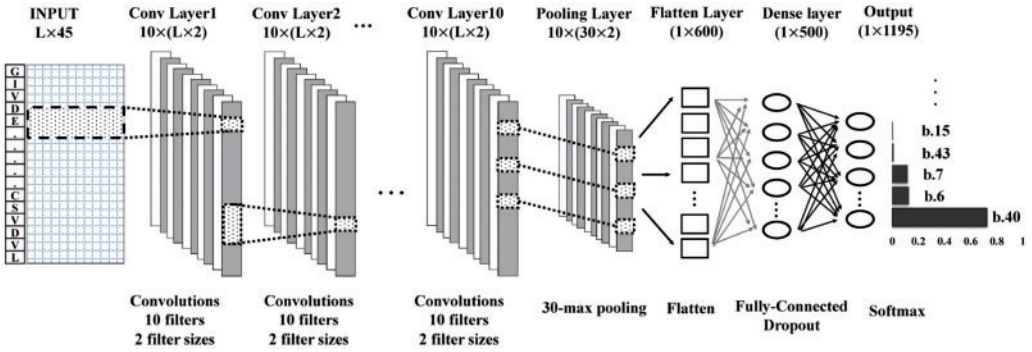

Fig. 1. The architecture of 1D deep convolutional neural network for fold classification. The network accepts the features of proteins of variable sequence length (L) as input, which are transformed into hidden features by 10 hidden layers of convolutions. Each convolution layer applies 10 filters to the windows of previous layers to generate L hidden features. Two window sizes (6 and 10) are used. The 30 maximum values of hidden values of each filter of the 10th convolution layer are selected by max pooling, which are joined together into one vector by flattening. The hidden features in this vector are fully connected to a hidden layer of 500 nodes, which are fully connected to 1195 output nodes to predict the probability of each of 1195 folds. The output node uses softmax function as activation function, whereas all the nodes in the other layers use rectified linear function (ReLU) as activation function. The features in the convolution layers are normalized by batches

with two window sizes (i.e. $10 \times L \times 2$) in the first convolution layer are as input to be transformed by the second convolution layer in the same way. The depth of convolution layers is set to 10. Inspired by the work (Kalchbrenner *et al.*, 2014), the K-max pooling layer is added to transform the hidden features of variable length in the last convolution layer to the fixed number of features, where K is set to 30. That is the 30 highest values (30 most active features) of each $L \times 1$ feature map generated by a filter with a window size are extracted and combined. The extracted features learned from both window sizes (i.e. 6, 10) are merged into one single vector consisting of $10 \times 30 \times 2$ numbers, which is fed into a fully-connected hidden layer consisting of with 500 nodes. These nodes are fully connected to 1195 nodes in the output layer to predict the probability of 1195 folds. The node in the output layer uses the softmax activation function. To prevent the over-fitting, the dropout (Srivastava *et al.*, 2014) technique is applied in the hidden layer (i.e. the 14th layer in Fig. 1).

2.4 Model training and validation

We trained the one-dimensional deep convolutional neural network (DeepSF) on variable-length sequences in 1195 folds. Considering the proteins in the training dataset have very different length of up to 1419 residues, we split the proteins into multiple mini-batches (bins) based on fixed-length interval (bin size). The proteins in the same bin have similar length in a specific range. The zero-padding is applied to the sequences whose length is smaller than the maximum length in the bin. All the mini-batches are trained for 100 epochs, and the proteins in each bin are used to train for a small number of epochs (i.e. 3 epochs for bin with size of 15) in order to avoid over-training on the proteins in a specific bin. We evaluated the performance of different bin sizes (see the Result section) to choose a good bin size. The DeepSF with different parameters is trained on the training dataset with less than or equal to 95% pairwise similarity, and is then evaluated on the validation sets with different sequence similarity levels (95, 70, 40, 25%) or at three hierarchical levels (family/superfamily/fold) with the training dataset. The model with the best average accuracy on the validation datasets is selected as final model for further testing and evaluation. A video demonstrating how DeepSF learns to classify a protein into a correct fold during training is available http://iris.rnet.missouri.edu/DeepSF/.

2.5 Model evaluation and benchmarking

We tested our method on the two independent test datasets: SCOP 2.06 (see Section 2.1.2) and CASP dataset (see Section 2.1.3). Since the number of proteins in different folds are extremely unbalanced, we split the 1195 folds into three groups based on the number of proteins within each fold (i.e. small, medium, large). A fold is defined as 'small' if the number of proteins in the fold is less than 5, 'medium' if the number of proteins is in the range between 6 and 50, and 'large' if the number of proteins is larger than 50. We evaluated DeepSF on the proteins of all folds and those of each category in the test dataset separately. We also compared DeepSF with a state-of-the-art profile–profile alignment method—HHSearch and PSI-BLAST on the CASP dataset based on top1, top5, top10 predictions, respectively.

2.6 Hidden fold-related feature extraction and template ranking

The outputs of the 14th layer of DeepSF (the hidden layer in fully connected layers) used to predict the folds can be considered as the hidden, fold-related features of an input protein, referred to as SF-Feature. The hidden features bridge between the protein sequence space and the protein fold space as the embedded word features connect a natural language sentence to its semantic meaning in natural language processing. Therefore, the hidden features extracted for proteins by DeepSF can be used to assess the similarity between proteins and can be used to rank template proteins for a target protein.

In our experiment, we evaluated the following four different distance (or similarity) metrics to measure the similarity between the fold-related features:

1. Euclidean distance:

$$\text{Euclid-D} : (Q, T) \mapsto \sqrt{\sum_{i=1}^{N} (Q_i - T_i)^2} \quad (1)$$

2. Manhattan distance:

$$\text{Manh-D} : (Q, T) \mapsto \sum_{i=1}^{N} |Q_i - T_i| \quad (2)$$

3. Pearson's Correlation score:

$$\text{Corr-D} : (Q, T) \mapsto \log(1 - \text{Corr}(Q, T)) \quad (3)$$

4. KL-Divergence:

$$\text{KL-D} : (Q, T) \mapsto \sum_{i=1}^{N} \left(\text{Q}_\text{i} \log \frac{\text{Q}_\text{i}}{\text{T}_\text{i}} + \text{T}_\text{i} \log \frac{\text{T}_\text{i}}{\text{Q}_\text{i}} \right) \quad (4)$$

where Q, T is the SF-feature for query protein and template protein.

We randomly sampled 5 folds from the training dataset and sampled at most 100 proteins from the 5 folds to test the four metrics above. We use hierarchical clustering to cluster the proteins into five clusters, where the distance between any two proteins is calculated from their fold-related feature vectors by the four metrics, respectively. This process is repeated 1000 times and the accuracy of clustering based on the four distance metrics are calculated and compared (see Results Section 3.4).

To select the best template for a target protein, the fold-related features of the target protein is compared with those of the proteins in the fold that the target protein is predicted to belong to. The templates in the fold are ranked in terms of their distance with the target protein.

3 Results

3.1 Training and validation on SCOP 1.75 dataset

We trained the deep convolutional neural network (DeepSF) on SCOP 1.75 dataset in the mini-batch mode, where the proteins in each mini-batch (bin) have similar length. We evaluated the effects of different bin sizes: 500, 200, 50, 30, 15 and each size ranging from 1 to 15. The numbers of proteins within each batch (bin) are visualized in Supplementary Figure S3. The classification accuracy on the validation dataset with different bin sizes for each epoch of training is shown in Supplementary Figure S4. Bin size of 15 has the fastest convergence and highest accuracy on both training (see Supplementary Fig. S4a) and validation datasets (see Supplementary Fig. S4b and c), and therefore is chosen taking both accuracy and running time (see Supplementary Table S7) into account. For the test dataset of SCOP 1.75, we evaluated the performance of DeepSF at family, superfamily and fold level against training datasets. As shown in Table 1, at the family level, DeepSF achieves the accuracy of 76.18% for top prediction, which is worse than a standard sequence alignment method—PSI-BLAST. At the superfamily level, for top 1 (or top 5) prediction, the accuracy of DeepSF is 50.71% (or 77.67%), which is much higher than 42.20% (or 51.40%) of PSI-BLAST. At the fold level, for top 1 (or top 5) prediction, the accuracy of DeepSF is 40.95% (or 70.47%), which is many times better than 5.60% (or 11.60%) of PSI-BLAST. It is worth noting that the accuracy of PSI-BLAST is calculated based on the top folds from the ranked templates. The results show that DeepSF recognizes folds much better than PSI-BLAST for hard cases when sequence identity is very low.

On the validation datasets whose redundancy is reduced to at most 95, 70, 40 and 25% sequence similarity with the training dataset, DeepSF achieves the accuracy of 80.4% (or 93.7%) for top 1 (or top 5) predictions at the 95% similarity level. The average accuracy on all the four validation datasets (95%/70%/40%/25%) is 75.3% (or 90.9%) for top 1 (or top 5) predictions. The detailed results on these validation datasets are reported in Supplementary Table S1.

Table 1. The prediction accuracy at family/superfamily/fold levels for top 1, top 5 and top 10 predictions of DeepSF and PSI-BLAST, on SCOP 1.75 test dataset

Level	Methods	Top1	Top5	Top10
Family (1272 proteins)	DeepSF	76.18%	94.50%	97.56%
	PSI-BLAST	96.80%	97.40%	97.60%
Superfamily (1254 proteins)	DeepSF	50.71%	77.67%	77.67%
	PSI-BLAST	42.20%	51.40%	54.60%
Fold (718 proteins)	DeepSF	40.95%	70.47%	82.45%
	PSI-BLAST	5.60%	11.60%	16.20%

Table 2. The accuracy of DeepSF on SCOP 2.06 dataset and its subsets

DeepSF	Top1	Top5	Top10
SCOP2.06 dataset	73.00%	90.25%	94.51%
'Large' folds	79.64%	94.87%	97.81%
'Medium' folds	74.16%	75.61%	76.06%
'Small' folds	67.93%	86.86%	94.74%

3.2 Performance on SCOP 2.06 dataset

We evaluated DeepSF on the independent SCOP 2.06 dataset, which contains 2533 proteins belonging to 550 folds. 60 folds with 1326 proteins are considered as 'Large' fold, 249 folds with 898 proteins as 'Medium' fold and 241 folds with 307 proteins as 'Small' fold. The classification accuracy of DeepSF on all the folds and each kind of fold is reported in Table 2. The accuracy on the entire dataset is 73.0 and 90.25% for top 1 prediction and top 5 predictions, respectively. The model also achieves accuracy of 79.64, 74.16 and 67.93% for top 1 prediction on 'Large', 'Medium' and 'Small' folds, respectively. The higher accuracy on larger folds suggests that more training data in a fold leads to the better prediction accuracy. The classification accuracy of DeepSF on SCOP 2.06 dataset at family, superfamily and fold level against training dataset is reported in Table 3.

3.3 Performance on CASP dataset

We evaluated our method on the CASP dataset, including 95 template-free proteins and 88 template-based proteins. We compared our method with the two widely used alignment methods (HHSearch and PSI-BLAST). Our method predicts the fold for each CASP target from its sequence directly. HHSearch and PSI-BLAST search each CASP target against the proteins in the training dataset to find the homologs to recognize its fold, where the accuracy of PSI-BLAST/HHSearch is calculated based on the top ranked folds from the identified templates. The method for assigning folds to CASP targets is described in Section 2 in the Supplementary Material.

As shown in the Tables 4 and 5, DeepSF achieved better accuracy on both template-based targets and template-free targets than HHSearch, PSI-BLAST in all situations. On the template-based targets that have little similarity with training proteins, the accuracy of DeepSF for top 1, 5, 10 predictions are 46.59, 73.86, 84.09% (see Table 4), which is 3.39, 12.46, 17.09% higher than HHSearch. And interestingly, the consensus ranking of HHSearch and DeepSF (Cons_HH_DeepSF) is better than both DeepSF and HHSearch, particularly for top 1 prediction, suggesting that the two methods are complementary on template-based targets. Because CASP targets have very low sequence similarity (<10%) with the training dataset, which is difficult for profile–sequence alignment methods to recognize, PSI-BLAST has the lowest prediction accuracy. On the hardest

Table 3. The prediction accuracy at family/superfamily/fold level for top 1, top 5 and top 10 predictions, on SCOP 2.06 test dataset

Type	Methods	Top1	Top5	Top10
Family (742 proteins)	DeepSF	75.87%	91.77%	95.14%
	PSI-BLAST	82.20%	84.50%	85.30%
Superfamily (1754 proteins)	DeepSF	72.23%	90.08%	94.70%
	PSI-BLAST	86.90%	88.40%	89.30%
Fold (37 proteins)	DeepSF	51.35%	67.57%	72.97%
	PSI-BLAST	18.90%	35.10%	35.10%

Table 4. The performance of the methods on 88 template-based proteins in the CASP dataset

Method	Top1	Top5	Top10
DeepSF	46.59%	73.86%	84.09%
HHSearch	43.20%	61.40%	67.00%
Cons_HH_DeepSF	59.10%	77.30%	85.20%
PSI-BLAST	15.90%	31.80%	47.70%

template-free targets that presumably have no sequence similarity with the training dataset, the accuracy of DeepSF for top 1, 5 and 10 predictions are 24.21, 51.58 and 70.53% (see Table 5), 12.63, 16.84 and 26.32% higher than HHSearch that performs better than PSI-BLAST. The consensus (Cons_HH_DeepSF) of DeepSF and HHSearch is only slightly better than DeepSF, which is different from its effect on template-based modeling targets.

3.4 Evaluation of four distance metrics for comparing fold-related hidden features

We evaluated the four distance metrics by using hierarchical clustering to cluster proteins with known folds based on their hidden fold-related features (see Materials and methods Section 2.6). The boxplot in Figure 2a shows the clustering accuracy of 4 different distance metrics. While Euclid-D, Manh-D and Corr-D achieve accuracy of 86.3, 80.4 and 88.0%, KL-D performs the best with accuracy of 89.3%. Figure 2b shows an example that using KL-D as distance metric to cluster the fold-level features of proteins in five SCOP2.06 folds that are randomly sampled. The proteins are perfectly clustered into 5 groups with the same folds. The visualized heat map (Fig. 2b) shows that proteins in the same cluster (fold) has the similar hidden feature values. More detailed information including the name and SCOP id of the proteins is illustrated in Supplementary Figure S5.

3.5 Fold-classification assisted protein structure prediction

Since applying a distance metric such as KL-D to the fold-related hidden features of two proteins can be used to measure their structural similarity, we explored the possibility of using it to rank template proteins for a target protein to assist tertiary structure prediction. Using the DeepSF model, we can generate fold-related features (SF-features) for any protein in a template protein database. In our experiment, we use DeepSF to generate SF-features for all the proteins in the training dataset as the template database. Given a target protein, we first extracted its SF-features and predicted the top 5 folds for it. We selected top 5 folds because top 5 predictions generally provided the high accuracy of fold prediction. Then we collected the template proteins that belong to the predicted top

Table 5. The performance of the methods on 95 template-free proteins in the CASP dataset

Method	Top1	Top5	Top10
DeepSF	24.21%	51.58%	70.53%
HHSearch	11.58%	34.74%	44.21%
Cons_HH_DeepSF	23.16%	56.84%	70.53%
PSI-BLAST	8.42%	15.79%	32.63%

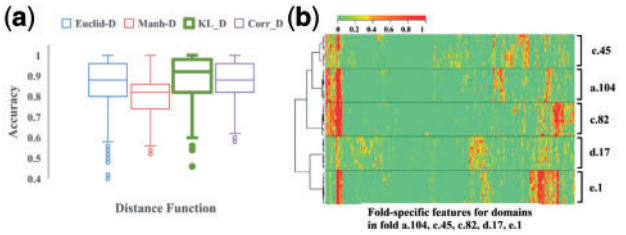

Fig. 2. (**a**) The accuracy of 4 distance metrics in clustering proteins based on fold-related features. The clustering accuracy is average over 1000 clustering processes. (**b**) A hierarchical clustering of proteins from 5 folds in the SCOP 2.06 dataset using KL-D as metric. Each row in the heat map visualizes a vector of fold-related hidden features of a protein. The feature vectors of the proteins of the same fold are similar and clustered into the same group

5 folds and compare their SF-features with that of the target protein using KL-D metric. The templates are then ranked by KL-D scores from smallest to largest, and the top ranked 10 templates are selected to build the protein structures for the target proteins (Cui *et al.*, 2016). This method contrasts with the approach of HHSearch, where the target sequence is searched against the template database, and the top ranked 10 templates with smallest e-value are selected as candidate templates for protein structure prediction.

After the templates are detected by DeepSF or HHSearch, the sequence alignment between the target protein and each template are generated using HHalign (Söding, 2005). Each alignment and its corresponding template structure are fed into Modeller (Webb and Sali, 2014) to build the tertiary structures. The predicted structural model with highest TMscore among all the models generated by top templates is selected for comparison. The quality of best predicted models from DeepSF and HHSearch is evaluated against the native structure in terms of TM-score and RMSD (Zhang and Skolnick, 2005).

Here, we mainly evaluated template ranking and protein structure prediction on the 95 template-free CASP targets assuming that our method is more useful for detecting structural similarity for hard targets without sequence similarity with known templates. Table 6 reports the average, min, max and standard deviation (std) of TMscore of the best models predicted for 95 template-free targets by DeepSF and HHSearch. DeepSF achieved a higher average TMscore (0.27) than that (0.25) of HHSearch. And the p-value of the difference using Wilcoxon paired test is 0.019.

Figure 3 shows an example on which DeepSF performed well. T0862-D1 is a template-free target in CASP 12, which contains multiple helices. DeepSF firstly classifies T0862-D1 into fold 'a.7' with probability 0.77 which is a 3-helix bundle. And among the top 10 ranked templates with smallest KL-D score in the fold 'a.7', the domain 'd1wr0a1' (SCOP id: a.7.14.1) was used to generate the best structural model with TMscore = 0.54 and RMSD = 4.6 Angstrom. In contrast, among the top 10 predicted structural models from HHSearch, the best model was constructed from a segment (residues

Table 6. Accuracy of protein structure predictions on 95 template-free targets

Methods	TM-score			
	Min	Max	Mean	Std
DeepSF	0.15	0.54	0.27	0.07
HHSearch	0.11	0.52	0.25	0.08

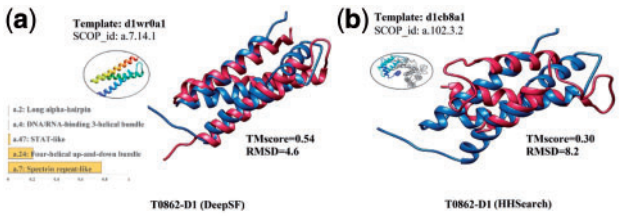

Fig. 3. Tertiary structure prediction for CASP12 target T0862-D1 based on templates identified by DeepSF and HHSearch. (**a**) DeepSF predictions: a top template, five predicted folds and the supposition between the best model and the template structure; (**b**) HHSearch predictions: top template, and superposition of the best model and the template structure

5–93) of a large template 'd1cb8a1' (SCOP id: a.102.3.2), which has TMscore of 0.30 and RMSD of 8.2.

3.6 Robustness of fold-related features against sequence mutation, insertion, deletion and truncation

In the evolutionary process of proteins, amino acid insertion, deletion or mutations mostly modifies protein sequences without changing the structural fold. Protein truncation that shortens the protein sequences at either N-terminal or C-terminal sometimes still retains the structural fold (Jackson and Fersht, 1991). A good method of extracting fold-related features from sequences should capture the consistent patterns despite of the evolutionary changes. Therefore, we simulated these four residue changes to check if the fold-related features extract from protein sequences by DeepSF are robust against mutation, insertion, deletion and even truncation. To analyze the effects of mutation, insertion and deletion, we selected some proteins that have 100 residues, and randomly selected the positions for insertion, deletion, or substitution with one or more residues randomly sampled from 20 standard amino acids. And at most 20 residues in total are deleted from or inserted into sequences. Each change was repeated 50 times, and the exactly same sequences were removed after sampling. For example, for domain d1lk3h2 we generated 44 sequences with at least one residue deleted, and 44 sequences with at least one residue insertion, and 18 sequences with at least one residue mutation. The SF-Features for these mutated sequences are generated and compared to the SF-Feature of the original wild-type sequence. We also randomly sampled 500 sequences with length in the range of 80–120 residues from the SCOP 1.75 dataset as control, and compare their SF-features with those of the original sequence. The distribution of KL-D divergences between the SF features of these sequences and the original sequence are shown in Figure 4. The divergence of the sequences with mutations, insertions and deletions from the original sequence is much smaller than that of random sequences. The p-value of difference according to Wilcoxon rank sum test is <2.2e-16. The same analysis is applied to the other two proteins: 'd1foka3' and 'd1ipaa2', and the same phenomena has been observed (see Supplementary Fig. S6). The results

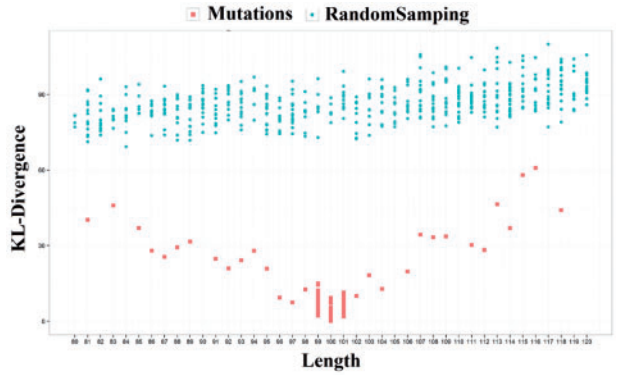

Fig. 4. The KL-D divergences of fold-related features of 106 modified sequences of protein d1lk3h2 from the wild-type sequence (red dots) and those of 500 random sequences from the wild-type sequence (blue dots) (Color version of this figure is available at *Bioinformatics* online.)

suggest that the feature extraction of DeepSF is robust against the perturbation of sequences.

For the truncation analysis, we simulated residue truncations on C-terminus of 4188 proteins in the SCOP 2.06 datasets (identity 40% against SCOP1.75) by letting DeepSF read each protein's sequence from N-terminal to C-terminal to predict its fold. DeepSF needs to read 67.1% of the original sequences from N- to C-terminal on average in order to predict the same fold as using the entire sequences. This may suggest that the feature extraction is robust against the truncation of residues at C-terminal. A video demonstrating how DeepSF reads a protein sequence from N- to C-terminal to predict fold is available at http://iris.rnet.missouri.edu/DeepSF/.

3.7 Generalization of deep convolutional neural network for family classification on SCOP database and fold classification on ECOD database

We generalized our method to the family level classification involving 3901 families in the SCOP1.75 database. On the test dataset, the prediction was 61.21% (or 79.42%) for top1 (or top 5) prediction. Detailed results are described in the Section 3 in the Supplementary Material. Moreover, we trained our method on the ECOD database (Cheng *et al.*, 2014), which is a hierarchical domain classification database based on the distant evolutionary relationships between proteins. We designed two architectures to classify 2186 possible homologous groups (sharing similar structure but lack a convincing argument for homology) with an accuracy of 50.95% (or 78.23%) for top 1 (or top 5) prediction and 3459 homologous groups with an accuracy of 47.46% (or 71.52%) for top 1 (or top 5) prediction. The detailed analysis of the results is reported in Section 4 in the Supplementary Material.

3.8 Analysis of the importance of the features for fold classification

In this study, four kinds of sequence and structure features were generated to represent a protein. It is worth analyzing their importance for fold classification. 15 different combinations of features were trained with 1D-convolutional neural network separately. The results on the SCOP 1.75 validation dataset are summarized in Supplementary Figure S19. Secondary structure features make most significant contributions to the fold classification, whose accuracy of top 1 prediction is at least 6.48% higher than the other three individual features. And combining all 4 features leads to the best

performance. Due to the significant effect of secondary structure, we analyzed how the different quality of predicted secondary structure influences the fold prediction. We generated predicted secondary structure using four methods: SCRATCH (Magnan and Baldi, 2014), DeepCNF (Wang *et al.*, 2015), DNSS (Spencer *et al.*, 2015) and PSIPRED (McGuffin *et al.*, 2000), which were used for fold classification on the CASP dataset, respectively. The results are shown in Supplementary Tables S8 and S9. For top 1 fold prediction, higher secondary structure prediction accuracy generally leads to higher fold classification accuracy. More details are described in the Supplementary Section S7.

4 Conclusion

We presented a deep convolution neural network to directly classify a protein sequence into one of all 1195 folds defined in SCOP 1.75. To our knowledge, this is the first system that can directly classify proteins from the sequence space to the entire fold space rather accurately without using sequence comparison. Our method can automatically extract a set of fold-related hidden features from protein sequence of any length by deep convolution, which is different from previous machine learning methods relying on a window of fixed size or human expertise for feature extraction. The automatically extracted features are robust against sequence perturbation and can be used for various protein data analysis such as protein comparison, clustering, template ranking and structure prediction. And on the independent test datasets, our method is more accurate in recognizing folds of target proteins that have little or no sequence similarity with the proteins having known structures than widely used profile–profile alignment methods. Moreover, our method of directly assigning a protein sequence to a fold is not only complementary with traditional sequence-alignment methods based on pairwise comparison, but also provides a new way to study the protein sequence–structure relationship.

Acknowledgements

The computation for this work was performed on the high performance computing infrastructure provided by Research Computing Support Services and in part by the National Science Foundation under grant number CNS-1429294 at the University of Missouri, Columbia Mo.

Funding

This work was supported by an National Institutes of Health (NIH) R01 grant (R01GM093123) to JC.

Conflict of Interest: none declared.

References

Altschul,S.F. *et al.* (1990) Basic local alignment search tool. *J. Mol. Biol.*, **215**, 403–410.

Altschul,S.F. *et al.* (1997) Gapped BLAST and PSI-BLAST: a new generation of protein database search programs. *Nucleic Acids Res.*, **25**, 3389–3402.

Berman,H.M. *et al.* (2000) The protein data bank. *Nucleic Acids Res.*, **28**, 235–242.

Cao,R. and Cheng,J. (2016) Integrated protein function prediction by mining function associations, sequences, and protein–protein and gene–gene interaction networks. *Methods*, **93**, 84–91.

Chandonia,J.-M. *et al.* (2016) SCOPe: manual Curation and artifact removal in the structural classification of proteinsextended database. *J. Mol. Biol.*, **429**, 348–355.

Cheng,H. *et al.* (2014) ECOD: an evolutionary classification of protein domains. *PLoS Computat. Biol.*, **10**, e1003926.

Cheng,J. and Baldi,P. (2006) A machine learning information retrieval approach to protein fold recognition. *Bioinformatics*, **22**, 1456–1463.

Chung,I.-F. *et al.* (2003) Recognition of structure classification of protein folding by NN and SVM hierarchical learning architecture. In: Artificial Neural Networks and Neural Information Processing—ICANN/ICONIP 2003, pp. 179–179.

Cui,X. *et al.* (2016) CMsearch: simultaneous exploration of protein sequence space and structure space improves not only protein homology detection but also protein structure prediction. *Bioinformatics*, **32**, i332–i340.

Damoulas,T. and Girolami,M.A. (2008) Probabilistic multi-class multi-kernel learning: on protein fold recognition and remote homology detection. *Bioinformatics*, **24**, 1264–1270.

Dill,K.A. *et al.* (2008) The protein folding problem. *Annu. Rev. Biophys.*, **37**, 289–316.

Dong,Q. *et al.* (2009) A new taxonomy-based protein fold recognition approach based on autocross-covariance transformation. *Bioinformatics*, **25**, 2655–2662.

Eickholt,J. and Cheng,J. (2012) Predicting protein residue–residue contacts using deep networks and boosting. *Bioinformatics*, **28**, 3066–3072.

Greene,L.H. *et al.* (2007) The CATH domain structure database: new protocols and classification levels give a more comprehensive resource for exploring evolution. *Nucleic Acids Res.*, **35**, D291–D297.

Hadley,C. and Jones,D.T. (1999) A systematic comparison of protein structure classifications: SCOP, CATH and FSSP. *Structure*, **7**, 1099–1112.

Henikoff,S. and Henikoff,J.G. (1992) Amino acid substitution matrices from protein blocks. *Proc. Natl. Acad. Sci. USA*, **89**, 10915–10919.

Holm,L. and Sander,C. (1994) The FSSP database of structurally aligned protein fold families. *Nucleic Acids Res.*, **22**, 3600.

Jackson,S.E. and Fersht,A.R. (1991) Folding of chymotrypsin inhibitor 2. 1. Evidence for a two-state transition. *Biochemistry*, **30**, 10428–10435.

Jo,T. and Cheng,J. (2014) Improving protein fold recognition by random forest. *BMC Bioinformatics*, **15**, S14.

Jo,T. *et al.* (2015) Improving protein fold recognition by deep learning networks. *Sci. Rep.*, **5**, 17573.

Kalchbrenner,N. *et al.* (2014) A convolutional neural network for modelling sentences. In *Proceedings of ACL*, pp. 655–665.

Kim,Y. (2014) Convolutional neural networks for sentence classification. In: *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*. Association for Computational Linguistics, pp. 1746–1751.

Kinch,L.N. *et al.* (2016) CASP 11 target classification. *Proteins Struct. Funct. Bioinform.*, **84**, 20–33.

Kinch,L.N. *et al.* (2011) CASP9 target classification. *Proteins Struct. Funct. Bioinform.*, **79**, 21–36.

Krizhevsky,A. *et al.* (2012) Imagenet classification with deep convolutional neural networks. In: *Advances in Neural Information Processing Systems*. pp. 1097–1105.

Li,W. and Godzik,A. (2006) Cd-hit: a fast program for clustering and comparing large sets of protein or nucleotide sequences. *Bioinformatics*, **22**, 1658–1659.

Ma,J. *et al.* (2014) MRFalign: protein homology detection through alignment of Markov random fields. *PLoS Comput. Biol.*, **10**, e1003500.

Magnan,C.N. and Baldi,P. (2014) SSpro/ACCpro 5: almost perfect prediction of protein secondary structure and relative solvent accessibility using profiles, machine learning and structural similarity. *Bioinformatics*, **30**, 2592–2597.

McGuffin,L.J. *et al.* (2000) The PSIPRED protein structure prediction server. *Bioinformatics*, **16**, 404–405.

Murzin,A.G. *et al.* (1995) SCOP: a structural classification of proteins database for the investigation of sequences and structures. *J. Mol. Biol.*, **247**, 536–540.

Shen,H.-B. and Chou,K.-C. (2006) Ensemble classifier for protein fold pattern recognition. *Bioinformatics*, **22**, 1717–1722.

Söding,J. (2005) Protein homology detection by HMM–HMM comparison. *Bioinformatics*, **21**, 951–960.

Spencer,M. *et al*. (2015) A deep learning network approach to ab initio protein secondary structure prediction. *IEEE/ACM Trans. Comput. Biol. Bioinform*., **12**, 103–112.

Srivastava,N. *et al*. (2014) Dropout: a simple way to prevent neural networks from overfitting. *J. Mach. Learn. Res*., **15**, 1929–1958.

Wang,S. *et al*. (2016) Protein secondary structure prediction using deep convolutional neural fields. *Sci. Rep*., **6**, 18962.

Wang,S. *et al*. (2017) Accurate de novo prediction of protein contact map by ultra-deep learning model. *PLOS Comput. Biol*., **13**, e1005324.

Wang,S. *et al*. (2015) DeepCNF-D: predicting protein order/disorder regions by weighted deep convolutional neural fields. *Int. J. Mol. Sci*., **16**, 17315–17330.

Webb,B. and Sali,A. (2014) Protein structure modeling with MODELLER. *Methods Mol Biol*., **1137**, 1–15.

Wei,L. *et al*. (2015) Enhanced protein fold prediction method through a novel feature extraction technique. *IEEE Trans. Nanobiosci*., **14**, 649–659.

Xia,J. *et al*. (2016) An ensemble approach to protein fold classification by integration of template-based assignment and support vector machine classifier. *Bioinformatics*, **33**, 863–870.

Xu,J. and Zhang,Y. (2010) How significant is a protein structure similarity with TM-score= 0.5?. *Bioinformatics*, **26**, 889–895.

Zhang,Y. and Skolnick,J. (2005) TM-align: a protein structure alignment algorithm based on the TM-score. *Nucleic Acids Res*., **33**, 2302–2309.