Biointerface Research in Applied Chemistry
Platinum Open Access Journal (ISSN: 2069-5837)

Article
Volume 10, Issue 5, 2020, 6306 - 6316
https://doi.org/10.33263/BRIAC105.63066316

Protein Fold Prediction for Protein Sequences of Low Identity Based on Evolutionary and Spatial Features Using Random Forest Algorithm

Apurva Mehta [1, *], Himanshu Mazumdar [2]

[1] Assistant Professor, Department of Computer Engineering, Faculty of Technology, Dharmsinh Desai Univeristy
[2] Head, R&D Center, Faculty of Technology, Dharmsinh Desai University
* Correspondence: apurvamehta.ce@ddu.ac.in;

Scopus Author ID 57212214531

Received: 23.04.2020; Revised: 10.05.2020; Accepted: 10.05.2020; Published: 13.05.2020

Abstract: Protein fold prediction is a milestone step towards predicting protein tertiary structure from protein sequence. It is considered one of the most researched topics in the area of Computational Biology. It has applications in the area of structural biology and medicines. Extracting sensitive features for prediction is a key step in protein fold prediction. The actionable features are extracted from keywords of sequence header and secondary structure representations of protein sequence. The keywords holding species information are used as features after verifying with uniref100 dataset using TaxId. Prominent patterns are identified experimentally based on the nature of protein structural class and protein fold. Global and native features are extracted capturing the nature of patterns experimentally. It is found that keywords based features have positive correlation with protein folds. Keywords indicating species are important for observing functional differences which help in guiding the prediction process. SCOPe 2.07 and EDD datasets are used. EDD is a benchmark dataset and SCOPe 2.07 is the latest and largest dataset holding astral protein sequences. The training set of SCOPe 2.07 is trained using 93 dimensional features vector using Random forest algorithm. The prediction results of SCOPe 2.07 test set reports the accuracy of better than 95%. The accuracy achieved on benchmark dataset EDD is better than 93%, which is best reported as per our knowledge.

Keywords: Evolutionary; Protein Fold; Protein secondary structure; Random Forest; Spatial; Structural classification of protein.

1. Introduction

Knowledge of protein folding provides vital insights on structural and functional aspects of proteins, in current times. Demystifying the process of protein three dimensional structure formations from protein sequence is among the most complex mysteries. Protein fold prediction is the intermediate stage in pipeline of protein tertiary structure discovery [1].

Essentially, protein folding leads a protein being transformed from its denatured state to its biologically and functionally active confirmation [2]. Protein fold prediction is acquiring three dimensional protein structures from protein sequences without being concerned of protein sequence similarity [3].

The protein fold prediction pipeline consists of two main stages. Feature extractions from protein sequence and modeling of features using machine learning algorithms [4]. There are several types of features that can be extracted from protein sequences like; sequential, physico-chemical, structural, functional and evolutionary [5,6]. Profile-profile sequence alignment technique is incorporated for predicting protein folds using concepts of hidden markov models [7,8]. This method helps in capturing evolutionary and remote homology information of protein sequences. The tri-gram technique, popularly used in natural language processing application is used for extracting features from PSSM [9]. Later, SVM is used for machine learning based modeling. Local evolutionary and predicted secondary structure based features are used with SVM algorithm for predicting protein fold for sequence of low identity [10][11]. Spatial separation based features from PSSM are extracted and used with SVM for protein fold prediction [12]. It considers amino acids that are not adjacent in the sequence. An ensemble classifier coupled with features of secondary structure, evolutionary information, functional domain information and physico-chemical properties is combined for protein fold recognition [13]. The results from work [14] suggest that structure based features may turnout significantly important for protein fold prediction.

This work is focused on using primary and secondary structure based features for protein sequence representation. The feature vector of 93-dimension is used with machine learning algorithm Random Forest. SCOPe 2.07 [15] dataset is used for protein folding data and EDD [16] is used as benchmark dataset.

2. Materials and Methods

2.1. Datasets.

The latest version of SCOPe [17] dataset SCOPe 2.07 is used for protein folding data. Protein sequences of low identity (40%) are considered in this work. As prediction performance reported for low identity sequences is yet require further improvements. The EDD dataset is used as benchmark dataset as it contains header information we use in this work for extracting species based features. The SCOPe 2.07 contains information on 1003 folds while EDD has information on 27 folds. This work focuses on the prediction of 27 folds present in both EDD and SCOPe 2.07 dataset. The details are shown in Table 1. It is clearly evident from Table 1, the data is highly imbalance.

Table 1. Summary of EDD and SCOPe 2.07 Datasets.

Class	**Folds**	**Number of samples with EDD**	**Number of samples with SCOPe 2.07**
α	a.1: Globin-like	41	58
α	a.3: Cytochrome c	35	40
α	a.4: DNA/RNA-binding 3-helical bundle	322	408
α	a.24: Four-helical up-and-down bundle	69	76
α	a.26: 4-helical cytokines	30	32
α	a.39: EF Hand-like	59	83
β	b.1: Immunoglobulin-like beta-sandwich	391	539
β	b.6: Cupredoxin-like	47	51
β	b.121:Nucleoplasmin-like/VP (viral coat and capsid proteins)	60	64
β	b.29: Concanavalin A-like lectins/glucanases	57	87
β	b.34: SH3-like barrel	129	173
β	b.40: OB-fold	156	183
β	b.42: beta-Trefoil	45	61
β	b.47: Trypsin-like serine proteases	45	54
β	b.60: Lipocalins	37	49
α/β	c.1: TIM beta/alpha-barrel	336	480
α/β	c.3: FAD/NAD(P)-binding domain	73	27
α/β	c.23: Flavodoxin-like	130	211
α/β	c.2: NAD(P)-binding Rossmann-fold domains	195	302
α/β	c.37: P-loop containing nucleoside triphosphate hydrolases	239	315

Class	**Folds**	**Number of samples with EDD**	**Number of samples with SCOPe 2.07**
α/β	c.47: Thioredoxin fold	111	204
α/β	c.55: Ribonuclease H-like motif	128	163
α/β	c.69: alpha/beta-Hydrolases	83	138
α/β	c.93: Periplasmic binding protein-like	16	92
α+β	d.15: beta-Grasp (ubiquitin-like)	121	148
α+β	d.58: Ferredoxin-like	339	438
G	g.3: Knottins(small inhibitors, toxins, lectins)	124	137

2.2. Data pre-processing.

It is understood from the work of [14] the feature extraction methods must include secondary structure representations. It is as protein folds due to arrangements of its secondary structure in space relative to one another. Secondary structure representation can be found from various prediction servers and tools [18,19]. Secondary structure representation is obtained from DSSP algorithm [20,21]. It uses 8 states 'G', 'H', 'I', 'E', 'T', 'B', 'S' and ' ' for representing secondary structure. For convenience purpose we use 'C' instead of ' '. Data pre-processing steps are performed as given in [14].

2.3. Technique of feature extraction.

Variation in protein structures is less in comparison to protein sequence variation [22]. Thus, protein folds are formed around various protein secondary structures' regularly repetitive patterns. Experimentally seven patterns are identified as features. These patterns are based on abundance of α compartments, β compartments, parallel β sheets, anti-parallel β sheets, helix loop, sheet loop and sandwiched sheet in helix confirmations. The patterns are formed using secondary structure 3 states representation. Protein secondary structure 3 states representation is obtained by labeling G, H and I states as α for helix, E and B as β for sheet and T, S and C as λ for a turn or unknown [23,24]. The conversion is performed to compare performance with past works as they have used 3 state representations. The global feature vector is constructed using Equation 1.

$$\varphi_1 = \begin{pmatrix} f(\beta),\ f(\alpha),\ f(\alpha\lambda_1\alpha\lambda_1\alpha),\ f(\beta\lambda_1\beta\lambda_1\beta), \\ f(\beta\lambda_1\beta),\ f(\beta\alpha\beta),\ f(\ \alpha\lambda_2\beta\lambda_2\alpha) \end{pmatrix} \quad (1)$$

Where, $\alpha = [\mathrm{G} \mid \mathrm{H} \mid \mathrm{I}]\{3,\},\ \beta = [\mathrm{E} \mid \mathrm{B}]\{2,\},\ \lambda_1 = [\mathrm{T} \mid \mathrm{S} \mid \mathrm{C}]^{+}, \lambda_2 = [\mathrm{T} \mid \mathrm{S} \mid \mathrm{C}]*$, f is a frequency function, {X,} indicates occurrences of X or more based on minimum pitch found in respective structure, + indicates one or more occurrences, * indicates zero or more occurrences, | denotes OR and G, H, I, E, T, B, S and C are 8 states of secondary structure.

The native (i.e. local) information can assist in distinguishing protein folds. It is noted after vigilant inspections of various protein folds structure that certain patterns are repeated in certain native spatial arrangements only adhering to functional activity. This native pool of information is included by splitting secondary structure representation into approximately four equal parts. Equation 1 is applied to each of four parts and shown in Equation 2.

$$\varphi_2 = \begin{pmatrix} f(\beta)_{p1}, f(\alpha)_{p1}, f(\alpha\lambda_1\alpha\lambda_1\alpha)_{p1}, f(\beta\lambda_1\beta\lambda_1\beta)_{p1}, \\ f(\beta\lambda_1\beta)_{p1}, f(\beta\alpha\beta)_{p1}, f(\alpha\lambda_2\beta\lambda_2\alpha)_{p1}, \\ f(\beta)_{p2}, f(\alpha)_{p2}, f(\alpha\lambda_1\alpha\lambda_1\alpha)_{p2}, f(\beta\lambda_1\beta\lambda_1\beta)_{p2}, \\ f(\beta\lambda_1\beta)_{p2}, f(\beta\alpha\beta)_{p2}, f(\alpha\lambda_2\beta\lambda_2\alpha)_{p2}, \\ f(\beta)_{p3}, f(\alpha)_{p3}, f(\alpha\lambda_1\alpha\lambda_1\alpha)_{p3}, f(\beta\lambda_1\beta\lambda_1\beta)_{p3}, \\ f(\beta\lambda_1\beta)_{p3}, f(\beta\alpha\beta)_{p3}, f(\alpha\lambda_2\beta\lambda_2\alpha)_{p3}, \\ f(\beta)_{p4}, f(\alpha)_{p4}, f(\alpha\lambda_1\alpha\lambda_1\alpha)_{p4}, f(\beta\lambda_1\beta\lambda_1\beta)_{p4}, \\ f(\beta\lambda_1\beta)_{p4}, f(\beta\alpha\beta)_{p4}, f(\alpha\lambda_2\beta\lambda_2\alpha)_{p4}, \end{pmatrix} \quad (2)$$

Where, $\alpha = [G \mid H \mid I]\{3,\}, \beta = [E \mid B]\{2,\}, \lambda_1 = [T \mid S \mid C]^+, \lambda_2 = [T \mid S \mid C]^*$, f is a frequency function of pattern, p_i indicates ith part of the sequence for 1≤i≤4, {X,} indicates occurrences of X or more based on minimum pitch found in respective structure, + indicates one or more occurrences, * indicates zero or more occurrences, | denotes OR and G, H, I, E, T, B, S and C are 8 states of secondary structure.

The frequency of global and local patterns exposes compositional, distributional and transitional information present within data. That serves its purpose in discriminating protein folds. Although, secondary structure based features have their own downside, when many patterns are part of a single sample. That happens frequently with classes α+β and α/β. These kinds of situations may lead to misclassification or correct classification with limited confidence. This is not enough for making machine learning algorithms robust.

To counter this problem, features based on primary structures are also included. Header information is part of SCOPe 2.07 and EDD datasets. Species information is included in headers of dataset. Uniref100 dataset also contains species information for respective protein sequences. The protein three dimensional structures are manifestation of protein function. The functional differences are observed in species. The difference in species information also leads to the difference in physico-chemical features like amino acid profile, surface charge, etc[25][26]. Thus, species are bound to have positive connection with protein folds. The species represent knowledge of evolutionary history of a protein sequence. The evolutionary knowledge strengthens the prediction of protein folds for sequence of low identity. The similarity of species information likely to observe in proteins categorized to similar fold (i.e. structure) and function in the future. This keywords based features when coupled with secondary structure based features provides a distinct way of distinguishing protein folds [14]. Steps to find importantly applicable and no repetitive keywords from header information of EDD and SCOPe 2.07 dataset are shown below in Algorithm 1 given in Table 2.

Table 2. Steps to find important keywords from header of dataset.

```
Algorithm 1: Protein sequence keyword extraction from protein sequences
of EDD, SCOPe 2.07
1.    For s = 1 to S
      1.1.  Extract header from each record s
      1.2.  Assign s to respective protein fold label l (i.e. l = 1 to L, where
            L=27)
2.    For l = 1 to L
      2.1.  Create a dictionary: DictO_l(keyword, frequency)
      2.2.  Sort dictionary DictO_l in ascending order of frequency
      2.3.  Remove (keyword, frequency) pair from the head (top 10%) and
            tail (bottom 10%) of DictO_l
3.    Merge DictO_l for l=1 to 27 into DictF after removing redundant
      keywords.
```

There are many keywords extracted using Algorithm 1. The keywords playing pivotal role are identified using Random forest algorithm for protein fold recognition. The gist of keyword is calculated based on their purity in given base classifier in comparison to other base classifiers built. Impurity calculation in python's scikit-learn is done using Gini importance [27][28]. The dictionary DictF has total 292 keywords for representing 27 protein folds. For reducing the sparseness of features, a set of 58 keywords are identified from 292 keywords with 85% support for representing 27 protein folds. Equation 3 shows feature vector based upon header information of the primary structure of protein.

The final feature set is of 93D. It is displayed in Equation 4. Equation 4 is used to create feature vector from protein sequences of low identity for predicting protein folds.

$$\varphi_3 = \left(\begin{array}{l} b(K_i) = 1 \\ \quad if \ K_i \in Header \ Information \ of \ Input \ Sequence \\ Otherwise \ \ 0 \end{array} \right) \qquad (3)$$

Where, b is a Boolean function determining the presence of keyword K in input sequence's header information for 1≤i≤58.

$$\varphi_f = (\varphi_1, \varphi_2, \varphi_3) \qquad (4)$$

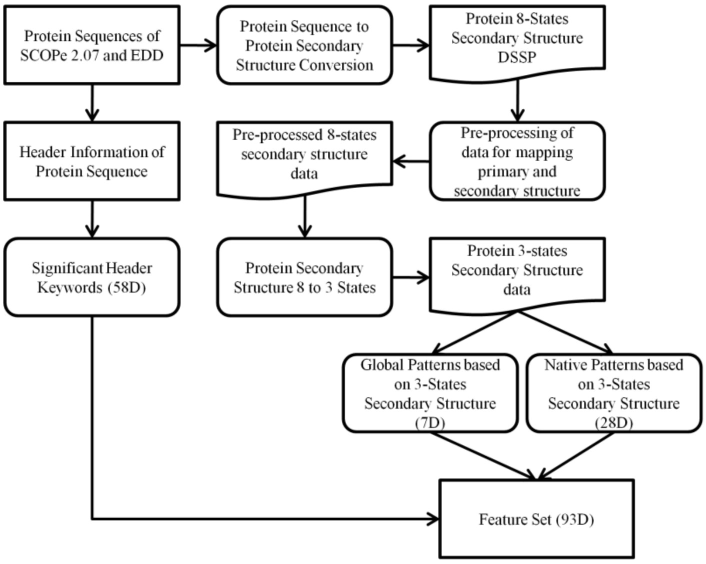
Protein Sequences of SCOPe 2.07 and EDD
Protein Sequence to Protein Secondary Structure Conversion
Protein 8-States Secondary Structure DSSP
Header Information of Protein Sequence
Pre-processed 8-states secondary structure data
Pre-processing of data for mapping primary and secondary structure
Significant Header Keywords (58D)
Protein Secondary Structure 8 to 3 States
Protein 3-states Secondary Structure data
Global Patterns based on 3-States Secondary Structure (7D)
Native Patterns based on 3-States Secondary Structure (28D)
Feature Set (93D)

Figure 1. Steps for Constructing Feature Vector from Protein Primary and Secondary Structure.

Figure 1 highlights the steps followed to construct feature vector of 93D using representation of primary and secondary protein structure.

2.4. Machine learning algorithm.

Protein fold prediction is targeted using various machine learning algorithms like SVM, ANN, tree classifiers, K-NN classifier, RF, Logistics tree, etc [5]. Recently, deep learning based methods are gaining popularity [29,30]. The current work selects RF [31] for model training and validation. The algorithm highlighting workings of RF is shown in Algorithm 2 provided in Table 3.

The RF is believed to be one among the robust ensemble algorithm [32]. As a first step bagging algorithm is used with original training data for creating training data with the notion of re-sampling. Then, the decision tree algorithm is used on bootstrapped data and arbitrarily selected features with aim of creating classifier of decision tree. Repeat this process for B times and create a collection of B decision trees. Sum up all the B decision trees and create concluding predictions based on averaging or majority voting mechanism. This divide and conquer principle of Random Forest algorithm is very effective in problems with large feature spaces as it focus on creating less correlated trees [33]. With the use of bagging in RF, it helps to create training data with diverse data. Even, not many hyper parameters to tune and computation complexity are of less degree in comparison to ANN and SVM. These all are advantageous in reducing overfitting during model training and creating a model more robust.

Table 3. Working of Random Forest algorithm.

```
Algorithm 2: RF  Algorithm
1.   For b = 1 to B
     1.1  Draw a bootstrap sample Z* from training data of size N
     1.2  Grow a random-forest tree $T_b$ to the bootstrapped data, by recursively repeating following steps for each terminal node of each tree, until the minimum node size $n_{min}$ is reached or any other termination criterion is reached.
          1.1.1.  Select m features at random from p features
          1.1.2.  Select the feature with best split point
          1.1.3.  Split the nodes into children nodes
2    Output the ensemble trees $\{T_b\}_1^B$
```

The scikit-learn library [27] is used for protein fold recognition. Bootstrap sampling is used in creating training datasets for each tree classifier in the forest. This bootstrap sampling may contain few samples more than once and few samples may not be included at all from original training set. Arbitrarily chosen features and class labels from training set corresponding to a classifier tree are used for model construction using Random Forest algorithm. The quality of split during tree construction is checked using criterion of Gini index. It follows three steps. It starts with checking impurity at node j using Equation 5. Afterwards, each attribute from arbitrarily chosen set is assessed for each of its value with the aim of reducing entropy after a split at node j using Equation 6. Then, impurity reduction after a split based on the chosen attribute is assessed using Equation 7. The attribute that gives maximum drop in entropy after split with Equation 7 is at last chosen for split at node j.

$$Gini(j) = 1 - \sum_{i=0}^{c-1} [p(i \mid j)]^2 \tag{5}$$

Where i represent possible class labels, is a function calculating conditional probability.

$$Gini_f(j) = \frac{|D1|}{|D|} Gini(D1) + \frac{|D2|}{|D|} Gini(D2) \tag{6}$$

Where f is a feature from random feature subset at node j, D indicates samples before node split, $D1$ and $D2$ samples created due to split.

$$\Delta Gini_f(j) = Gini(j) - Gini_f(j) \tag{7}$$

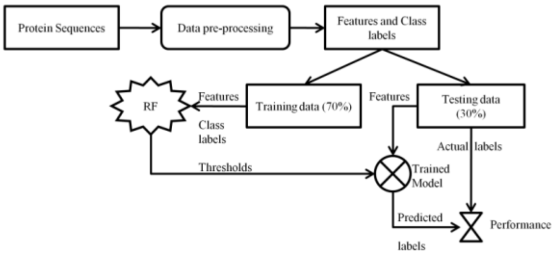

Figure 2. Machine Learning Pipeline for Training and Validating Protein Folds using RF Algorithm.

Figure 2 shows machine learning pipeline followed for training and validation of protein folding prediction. It initiates at protein sequences. As mentioned earlier data-preprocessing and feature extraction are performed. Dataset of features and class labels is partitioned into training data and testing data arbitrarily. Training data and testing data are consisting of 70% and 30% of original records respectively. Thresholds for building trained model are found based on trained classifier. Features from test data are used for predicting protein folds from the trained model and compared with an actual class for performance evaluation.

3. Results and Discussion

The results of the proposed approach are obtained with two datasets EDD and SCOPe 2.07. The significance of species based feature is highlighted by measuring performance of classifier supported with secondary structure based feature vectors.

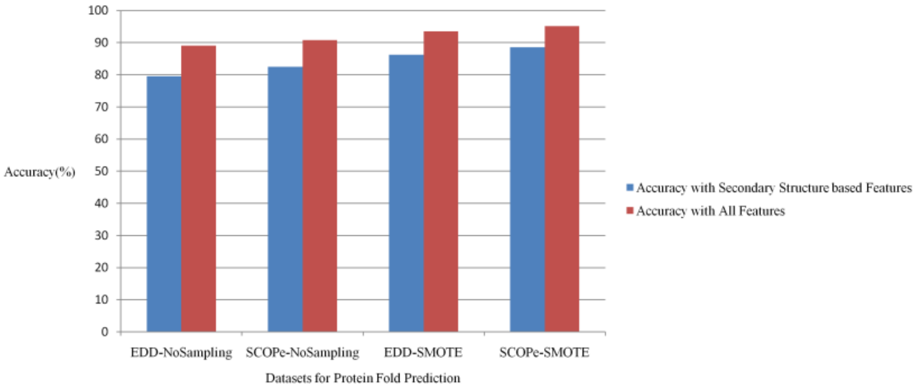

Figure 3. Performance Comparison for Protein Fold Recognition with Different Features and Sampling technique.

Figure 3 shows the performance with secondary structure based features and all features combined. It is observed with the inclusion of species based features accuracy goes up approximately 8-10%. Imbalance data problem is handled by using sampling technique SMOTE [34,35]. The SMOTE technique cares for imbalance data by over sampling minority labels and under sampling the majority labels. The impact due to usage of SMOTE in performance is also depicted in Figure 3. It is observed, EDD and SCOPe perform better with SMOTE sampling and all features used together. This shows the strength of the model has enhanced with the consideration of species based features.

Table 4 shows performance in terms of accuracy, precision and recall for each fold of EDD and SCOPe 2.07 datasets. Accuracy is just the ratio of correctly predicted samples to all available samples. Accuracy, informs us about training efficiency of the model and how it may score for future samples. It is alone inappropriate to use when you have highly imbalance data, like in this work. Precision and recall are much necessary for assessing the performance of imbalance data. Precision is the ratio of truly predicted positive samples to all predicted true samples. Precision helps counter costs associated with false positives. Recall is the ratio of truly predicted positive samples to all samples belonging in actual class. Recall guide us in handling costs attached to false negatives.

Table 4 includes performance information with SMOTE sampling. Following observations are made form Table 4.

- Irrespective of protein structural classes α, β, α+β and α/β most of the folds has good results with all three performances measurement.
- Performance for SCOPe 2.07 dataset is comparably higher than EDD dataset is largely due to availability of more observations as shown in Table 1.
- Misclassifications do occur but it is intra class, not inter class that suggest the strength of model being trained.
- Performance is approximately similar for folds belonging to respective protein structural classes, shows that model is not overfitting.

Figure 4 compares performance achieved by current work with major past works for protein fold prediction using EDD datasets. It is concluded from Figure 4 with benchmark datasets EDD current proposed approach achieves better results in comparison to recently published works in the area of protein fold prediction.

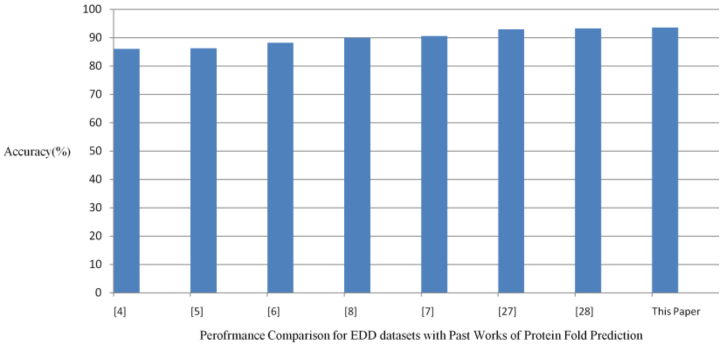

Figure. 4. Performance Comparison for Benchmark dataset EDD between Past works and Current work.

Table 4. Fold-wise performance comparison for EDD and SCOPe datasets.

	EDD-SMOTE			SCOPe 2.07-SMOTE		
Fold	**Accuracy**	**Precision**	**Recall**	**Accuracy**	**Precision**	**Recall**
a.1	100.00	100.00	100.00	100.00	100.00	100.00
a.3	100.00	100.00	100.00	100.00	100.00	100.00
a.4	100.00	94.92	100.00	96.88	100.00	96.88
a.24	100.00	100.00	100.00	100.00	83.33	100.00
a.26	100.00	88.89	100.00	66.67	100.00	66.67
a.39	89.47	100.00	89.47	100.00	100.00	100.00
b.1	93.90	91.67	93.90	98.21	96.49	98.21
b.6	53.85	100.00	53.85	40.00	100.00	40.00
b.121	90.91	90.91	90.91	87.50	100.00	87.50
b.29	92.86	76.47	92.86	100.00	71.43	100.00
b.34	85.71	85.71	85.71	95.45	100.00	95.45
b.40	90.24	94.87	90.24	100.00	84.62	100.00
b.42	100.00	100.00	100.00	100.00	85.71	100.00
b.47	100.00	100.00	100.00	100.00	100.00	100.00
b.60	100.00	88.90	100.00	100.00	100.00	100.00
c.1	96.55	98.25	96.55	96.43	96.43	96.43
c.3	93.33	93.33	93.33	100.00	95.00	100.00
c.23	100.00	75.00	100.00	100.00	100.00	100.00
c.2	100.00	100.00	100.00	71.43	100.00	71.43
c.37	97.37	90.24	97.37	100.00	94.12	100.00
c.47	87.50	87.50	87.50	100.00	90.00	100.00
c.55	86.36	86.36	86.36	100.00	100.00	100.00
c.69	95.83	92.00	95.83	100.00	90.91	100.00
c.93	100.00	100.00	100.00	100.00	100.00	100.00
d.15	100.00	95.65	100.00	100.00	100.00	100.00
d.58	88.89	96.55	88.89	96.55	93.33	96.55
g.3	96.77	100.00	96.77	84.62	100.00	84.62

4. Conclusions

In this work a Random forest based prediction model is build to predict 27 protein folds. Prediction model relies on 93 dimensional sound features vector. Features vector comprising of spatial and evolutionary information. Spatial information is taken from secondary structure representation and evolutionary information is captured from species details available in header of protein sequences. The structural representation highlights compositional, distributional, transitional, spatial aspects of sequence globally and locally. The species shows the functional aspects of sequence. Together they help build a good prediction model. The classifier achieves accuracy better than 95% with SCOPe 2.07 dataset. The classifier also reports accuracy better than 93% for benchmark dataset, which is best as per our knowledge. The current research work can be extended to predict protein folds from SCOPe 2.07 which are not included in this work. Even, protein structural class prediction results can be included as a feature while predicting protein fold, this will give an edge for improving prediction results further. Prediction algorithm can be modified to predict multiple possible folds for a protein sequence. Protein teriary structure modeling is possible with the knowledge of protein folds.

Funding

This research received no external funding.

Acknowledgments

Authors are thankful for Research and Development Center of Dharmsinh Desai University for allowing use of their tools and resources necessary for current research work.

Conflicts of Interest

The authors declare no conflict of interest.

References

1. Hou, J.; Adhikari, B.; Cheng, J. DeepSF: deep convolutional neural network for mapping protein sequences to folds. *Bioinformatics* **2018**, *34*, 1295–1303, https://doi.org/10.1093/bioinformatics/btx780.
2. Guo, J.; Rao, N.; Liu, G.; Yang, Y.; Wang, G. Predicting protein folding rates using the concept of Chou's pseudo amino acid composition. *J. Comput. Chem.* **2011**, *32*, 1612–1617, https://doi.org/10.1002/jcc.21740.
3. Ding, C.H.Q.; Dubchak, I. Multi-class protein fold recognition using support vector machines and neural networks. *Bioinformatics* **2001**, *17*, 349–358, doi:10.1093/bioinformatics/17.4.349, https://doi.org/10.1093/bioinformatics/17.4.349.
4. Yan, K.; Fang, X.; Xu, Y.; Liu, B. Protein Fold Recognition based on Multi-view Modeling. *Bioinformatics* **2019**, https://doi.org/10.1093/bioinformatics/btz040.
5. Wei, L.; Zou, Q. Recent Progress in Machine Learning-Based Methods for Protein Fold Recognition. **2016**, 1–13, https://doi.org/10.3390/ijms17122118.
6. Krishnan, S.M. Using Chou's general PseAAC to analyze the evolutionary relationship of receptor associated proteins (RAP) with various folding patterns of protein domains. *J. Theor. Biol.* **2018**, *445*, 62–74, https://doi.org/10.1016/j.jtbi.2018.02.008.
7. Lyons, J.; Dehzangi, A.; Heffernan, R.; Yang, Y.; Zhou, Y.; Sharma, A.; Paliwal, K. Advancing the accuracy of protein fold recognition by utilizing profiles from hidden Markov models. *IEEE Trans. Nanobioscience* **2015**, *14*, 761–772, https://doi.org/10.1109/TNB.2015.2457906.
8. Zhang, C.; Zheng, W.; Mortuza, S.M.; Li, Y.; Zhang, Y. DeepMSA: constructing deep multiple sequence alignment to improve contact prediction and fold-recognition for distant-homology proteins. *Bioinformatics* **2020**, *36*, 2105–2112, https://doi.org/10.1093/bioinformatics/btz863.

9. Paliwal, K.K.; Sharma, A.; Lyons, J.; Dehzangi, A. A Tri-Gram Based Feature Extraction Technique Using Linear Probabilities of Position Specific Scoring Matrix for Protein Fold Recognition. *NanoBioscience, IEEE Trans.* **2014**, *13*, 44–50, https://doi.org/10.1109/TNB.2013.2296050.
10. Dehzangi, A.; Paliwal, K.; Lyons, J.; Sharma, A.; Sattar, A. A segmentation-based method to extract structural and evolutionary features for protein fold recognition. *IEEE/ACM Trans. Comput. Biol. Bioinforma.* **2014**, *11*, 510–519, https://doi.org/10.1109/TCBB.2013.2296317.
11. Paliwal, K.K.; Sharma, A.; Lyons, J.; Dehzangi, A. Improving protein fold recognition using the amalgamation of evolutionary-based and structural based information. *BMC Bioinformatics* **2014**, *15*, S12, https://doi.org/10.1186/1471-2105-15-S16-S12.
12. Saini, H.; Raicar, G.; Sharma, A.; Lal, S.; Dehzangi, A.; Lyons, J.; Paliwal, K.K.; Imoto, S.; Miyano, S. Probabilistic expression of spatially varied amino acid dimers into general form of Chou′ s pseudo amino acid composition for protein fold recognition. *J. Theor. Biol.* **2015**, *380*, 291–298, https://doi.org/10.1016/j.jtbi.2015.05.030.
13. Chen, D.; Tian, X.; Zhou, B.; Gao, J. ProFold : Protein Fold Classification with Additional Structural Features and a Novel Ensemble Classifier. **2016**, *2016*, 26–33, https://doi.org/10.1155/2016/6802832.
14. Apurva, M.; Mazumdar, H. Predicting structural class for protein sequences of 40% identity based on features of primary and secondary structure using Random Forest algorithm. *Comput. Biol. Chem.* **2019**, 107164, https://doi.org/10.1016/j.compbiolchem.2019.107164.
15. Chandonia, J.-M.; Fox, N.K.; Brenner, S.E. SCOPe: classification of large macromolecular structures in the structural classification of proteins—extended database. *Nucleic Acids Res.* **2018**, *47*, D475--D481, https://doi.org/10.1093/nar/gky1134.
16. Dong, Q.; Zhou, S.; Guan, J. A new taxonomy-based protein fold recognition approach based on autocross-covariance transformation. **2009**, *25*, 2655–2662, https://doi.org/10.1093/bioinformatics/btp500.
17. Fox, N.K.; Brenner, S.E.; Chandonia, J.-M. SCOPe: Structural Classification of Proteins—extended, integrating SCOP and ASTRAL data and classification of new structures. *Nucleic Acids Res.* **2013**, *42*, D304--D309, https://doi.org/10.1093/nar/gkt1240.
18. Zhang, B.; Li, J.; Lü, Q. Prediction of 8-state protein secondary structures by a novel deep learning architecture. *BMC Bioinformatics* **2018**, *19*, 293, https://doi.org/10.1186/s12859-018-2280-5.
19. Kashani-Amin, E.; Tabatabaei-Malazy, O.; Sakhteman, A.; Larijani, B.; Ebrahim-Habibi, A. A systematic review on popularity, application and characteristics of protein secondary structure prediction tools. *Curr. Drug Discov. Technol.* **2019**, *16*, 159–172, https://doi.org/10.2174/1570163815666180227162157.
20. Kabsch, W.; Sander, C. Dictionary of protein secondary structure: pattern recognition of hydrogen-bonded and geometrical features. *Biopolym. Orig. Res. Biomol.* **1983**, *22*, 2577–2637, https://doi.org/10.1002/bip.360221211.
21. Touw, W.G.; Baakman, C.; Black, J.; te Beek, T.A.H.; Krieger, E.; Joosten, R.P.; Vriend, G. A series of PDB-related databanks for everyday needs. *Nucleic Acids Res.* **2014**, *43*, D364--D368, https://doi.org/10.1093/nar/gku1028.
22. Chothia, C. One thousand families for the molecular biologist. *Nature* **1992**, *357*, 543–544, https://doi.org/10.1038/357543a0.
23. Rost, B.; Eyrich, V.A. EVA: large-scale analysis of secondary structure prediction. *Proteins Struct. Funct. Bioinforma.* **2001**, *45*, 192–199, https://doi.org/10.1002/prot.10051.
24. Eyrich, V.A.; Mart\i-Renom, M.A.; Przybylski, D.; Madhusudhan, M.S.; Fiser, A.; Pazos, F.; Valencia, A.; Sali, A.; Rost, B. EVA: continuous automatic evaluation of protein structure prediction servers. *Bioinformatics* **2001**, *17*, 1242–1243, https://doi.org/10.1093/bioinformatics/17.12.1242.
25. Ikuma, K.; Shi, Z.; Walker, A. V; Lau, B.L.T. Effects of protein species and surface physicochemical features on the deposition of nanoparticles onto protein-coated planar surfaces. *RSC Adv.* **2016**, *6*, 75491–75498, https://doi.org/10.1039/C6RA13508K.
26. Steffen, P.; Kwiatkowski, M.; Robertson, W.D.; Zarrine-Afsar, A.; Deterra, D.; Richter, V.; Schlüter, H. Protein species as diagnostic markers. *J. Proteomics* **2016**, *134*, 5–18, https://doi.org/10.1016/j.jprot.2015.12.015.
27. Pedregosa, F.; Varoquaux, G.; Gramfort, A.; Michel, V.; Thirion, B.; Grisel, O.; Blondel, M.; Prettenhofer, P.; Weiss, R.; Dubourg, V.; et al. Scikit-learn: Machine learning in Python. *J. Mach. Learn. Res.* **2011**, *12*, 2825–2830.
28. Garreta, R.; Moncecchi, G. *Learning scikit-learn: machine learning in python*; Packt Publishing Ltd, 2013;
29. Liu, B.; Li, C.-C.; Yan, K. DeepSVM-fold: protein fold recognition by combining support vector machines

and pairwise sequence similarity scores generated by deep learning networks. *Brief. Bioinform.* **2019,** https://doi.org/10.1093/bib/bbz098.

30. Li, C.-C.; Liu, B. MotifCNN-fold: Protein fold recognition based on fold-specific features extracted by motif-based convolutional neural networks. *Brief. Bioinform.* **2019,** https://doi.org/10.1093/bib/bbz133.
31. Breiman, L. Random forests. *Mach. Learn.* **2001**, *45*, 5–32, https://doi.org/10.1023/A:1010933404324.
32. Liaw, A.; Wiener, M.; others Classification and regression by randomForest. *R news* **2002**, *2*, 18–22.
33. Biau, G.; Scornet, E. A random forest guided tour. *Test* **2016**, *25*, 197–227, https://doi.org/10.1007/s11749-016-0481-7.
34. Chawla, N. V; Bowyer, K.W.; Hall, L.O.; Kegelmeyer, W.P. SMOTE: synthetic minority over-sampling technique. *J. Artif. Intell. Res.* **2002**, *16*, 321–357, https://doi.org/10.1613/jair.953.
35. Fernández, A.; Garcia, S.; Herrera, F.; Chawla, N. V SMOTE for learning from imbalanced data: progress and challenges, marking the 15-year anniversary. *J. Artif. Intell. Res.* **2018**, *61*, 863–905, https://doi.org/10.1613/jair.1.11192.