5-2018

Prediction of DNA-Binding Proteins and their Binding Sites

Pujan Pokhrel
University of New Orleans

Prediction of DNA-Binding Proteins and their Binding Sites

An Honors Thesis

Presented to

the Department of Computer Science

of the University of New Orleans

In Partial Fulfillment

of the Requirements for the Degree of

Bachelor of Science, with University High Honors

and Honors in Computer Science

by

Pujan Pokhrel

May 2018

Acknowledgements

First and foremost, I am grateful to all the entities, known and unknown, that constitute our universe and to all those who constitute mine. I would like to thank my parents and my family for always being there for me.

I would like to thank Dr. Christopher Summa for introducing me to bioinformatics at my freshman year. I am grateful for the discussions with Aaron Maus, Bijay Regmi and Johnathan Redmann with whom I had the opportunity to learn from and share my knowledge.

To Dr. Tamjidul Hoque and Avdesh Mishra, thank you very much for introducing me to the problems in bioinformatics which can be solved using machine learning and helping me to hone my rudimentary approaches to science. I was a guy obsessed with Differentiable Neural Computers and deep learning during my sophomore year and wanted to use it on every problem I encountered. I am thankful to Dr. Sumaiya Iqbal with whom I first encountered the “No Free Lunch” theorem while working on evolutionary algorithms. I learned that every problem is unique and there exists no universal general-purpose algorithm that works for everything, not till now at least. Even nature is constantly optimizing the resources it has got, driven by entropy.

I would also like to acknowledge helpful discussions with Biplov Karkee, Manish Bhatt, Kauser Mohammed Ahmed, Abhishek Sapkota and Reecha Khanal.

Table of Contents

List of Tables

List of Figures

Abstract

DNA-binding proteins play an important role in various essential biological processes such as DNA replication, recombination, repair, gene transcription, and expression. The identification of DNA-binding proteins and the residues involved in the contacts is important for understanding the DNA-binding mechanism in proteins. Moreover, it has been reported in the literature that the mutations of some DNA-binding residues on proteins are associated with some diseases. The identification of these proteins and their binding mechanism generally require experimental techniques, which makes large scale study extremely difficult. Thus, the prediction of DNA-binding proteins and their binding sites from sequences alone is one of the most challenging problems in the field of genome annotation. Since the start of the human genome project, many attempts have been made to solve the problem with different approaches, but the accuracy of these methods is still not suitable to do large scale annotation of proteins. Rather than relying solely on the existing machine learning techniques, I sought to combine those using novel “stacking technique” and used the problem-specific architectures to solve the problem with better accuracy than the existing methods. This thesis presents a possible solution to the DNA-binding proteins prediction problem which performs better than the state-of-the-art approaches.

Keywords: Machine Learning, Large-Scale Data Analysis, Bioinformatics, DNA-Binding Proteins, DNA-Binding Residue.

Introduction

1.1 Thesis Overview

With the exponential growth of proteomic data and the enormous complexities involved in their modeling, bioinformatics becomes essential for the management and mining of biological data in modern biology, medicine and drug discovery. The development of computational tools to solve the problem requires the expertise from many fields of computer science, like i) Data science for mining, collection and preparation of data, ii) Scientific Computing to extract useful knowledge from large sets of data and mathematically quantify the knowledge as characteristics features, iii) Machine Learning to develop novel algorithms to model the data using features, and iv) Statistical and Probabilistic Analysis to empirically evaluate the model by comparative analysis and visualize the outputs. These methods have been utilized throughout the course of the thesis to develop novel tools for the prediction of DNA-binding proteins and their binding sites using sequence information only.

1.2 Contribution of the Thesis

Given the primary sequence of a protein as input, prediction of protein function is important for large scale annotations of various protein sequences gained from various large-scale genome sequencing projects. This thesis aims to solve the problem partially by providing a framework for the annotation of proteins based on their DNA-binding affinity. This work presents a framework for predicting if the protein is DNA-binding or not and if it is then what are its binding sites. The predictors developed in this work outperform all the other state-of-the-art predictors by a huge margin. The main aim of this thesis is to serve as a stepping stone for other similar predictions of protein functions and the use of complex stacked learners based architecture to solve machine learning problems.

1.3 Technical Results of the Thesis

- A stacked predictor for the prediction of DNA-binding proteins from sequences

This work presents a predictor, called StackDPPred, for the prediction of DNA-binding proteins using only the sequence information. In this work, I have developed a new stacking based prediction framework that utilizes a pool of base learners and a Support Vector Machine (SVM) on the meta layer to perform better predictions. Although the framework is developed for DNA-binding proteins, it can easily be tuned for other predictions. The predictor was developed using Python and is available online as both standalone code and as a web server. The performance of the predictor has been measured by employing various rigorous evaluation metrics. The results manifest that StackDPPred provides well-balanced and biologically relevant outputs for proteins of different lengths and with a wide variety of DNA-binding sites.

- A SVM based predictor for prediction of DNA-binding residues

In addition to DNA-binding proteins predictor, this work presents a predictor for the prediction of DNA-binding sites in the DNA-binding proteins using only the sequence information. Various useful features such as predicted disorder from DisPredict [1] along with other relevant features like evolutionary profile, monograms, bigrams and predicted torsion angles which helps improve the prediction accuracy are explored. In this work, Support Vector Machine (SVM) prediction framework has been proposed, which performs better than the state-of-the-art approaches. The performance of our predictor has been measured through standard evaluation metrics and through case studies. The results from our study shows that the predictor is well balanced on all the performance metrics and provides biologically relevant prediction of DNA-binding sites.

1.4 Thesis Organization

The main goal of this thesis is to develop computational tools for the prediction of DNA-binding proteins and their binding sites using only the sequence information. The rest of the thesis is organized as follows: Chapter 2 discusses the design and development of StackDPPred, a predictor of DNA-binding proteins. The framework is based on a pool of machine learning methods that are stacked together for superior performance. Chapter 3 describes the design and development of a predictor of DNA-binding sites in proteins. The framework is based on a Support Vector Machine (SVM) based architecture and is found to significantly outperform the state-of-the-art methods. Finally, Chapter 4 concludes this work, states the major contributions and provides brief future directions for further research to make the tools as accurate as possible.

1.5 Related Publications

Parts of this thesis work have been presented as poster and submitted to the Oxford journal of Bioinformatics for publication. The rest of the work is under preparation for publication. Below is the list of the publications:

1. Pujan Pokhrel, Avdesh Mishra and Md Tamjidul Hoque, "StackDPPred: Stacking based prediction of DNA-binding proteins from sequences." 6th Annual LA Conference on Computational Biology and Bioinformatics. 2018. (poster)
2. Avdesh Mishra, Pujan Pokhrel and Md Tamjidul Hoque, "StackDPPred: A stacking based prediction of DNA-binding proteins from sequences." Oxford Bioinformatics. (submitted)
3. Pujan Pokhrel, Avdesh Mishra and Md Tamjidul Hoque, "Prediction of DNA-binding sites from sequences using support vector machines." (preparing for submission)

Chapter 2

StackDPPred: A Stacking based Prediction of DNA-binding Protein from Sequences

2.1. Introduction

DNA-binding proteins carry out many crucial intercellular and intracellular functions such as DNA replication and repair, transcriptional regulation, the combination and separation of single-stranded DNA and other biological activities associated with DNA. In the eukaryotic cells, histone, which is a usual type of DNA-binding protein, often helps in packaging chromosomal DNA into a compact structure. Similarly, DNA-binding proteins, such as restriction enzymes, are DNA-cutting enzymes, which are found in bacteria that recognize and cut DNA only at a particular sequence of nucleotides to serve a host-defense role. DNA-binding proteins represent a broad category of proteins, which are found to be highly diverse in structure as well as in sequence. Structurally, they have been divided into 8 structural groups, which are further classified into 54 structural families [2, 3]. Additionally, it has been reported that 2 to 3% of a prokaryotic genomes and 6 to 7% of a eukaryotic genomes encode DNA-binding proteins [2, 4]. Moreover, the past decade has witnessed tremendous progress in genome sequencing [5-8]. According to the genome online database, the complete sequenced genomes of almost 1000 cellular organisms have been released, and about 5000 active genome sequencing projects are on the way [9, 10]. The unprecedented amount of genetic information has provided a plethora of protein sequences [11], posing a challenging problem of annotating them and elucidating their functions. Thus, the task of identifying DNA-binding proteins has become crucial.

In the early days, the identification of DNA-binding proteins was carried out by experimental techniques, including filter binding assays, genetic analysis, chromatin immunoprecipitation on microarrays and X-ray crystallography. However, it is both time-consuming and expensive to identify DNA-binding proteins purely based on biochemical experiments. Particularly, given the abundance of biological sequences generated in the post-genomic era. Hence, it is important to develop computational methods for fast and effective identification of DNA-binding proteins.

The computational methods refer to a wide range of approaches that capture various information such as structural, sequence and other physicochemical properties. Many attempts have been made in identifying DNA-binding proteins and many effective computational prediction methods have been proposed in the literature for analyzing them. The computational approaches for the identification of DNA-binding proteins can be divided into two broad categories: *i)* template based; and *ii)* machine learning based. Template-based methods can be further classified into two classes, one of which utilizes a structural comparison protocol and the other employs a sequence comparison protocol. The structural comparison protocol detects significant structural similarity between the query and a template known to bind DNA at either the domain or the structural motif, to assess the DNA-binding preference of the target sequence [12, 13]. Likewise, sequence comparison protocol (such as PSI-BLAST) detects significant sequence similarity between the query and a template known to bind DNA, to evaluate the DNA-binding preference of the target sequence [14]. Unlike template-based methods, machine learning methods do not employ direct structural comparison, rather they learn the relevant predictive model to make predictions by finding a pattern in the input feature space. Various machine learning algorithms such as support vector machine (SVM) [15-18], neural network [19-23], random forest [24], naïve Bayes classifier [25, 26], nearest neighbors algorithm [27] and ensemble classifiers [28-30] have been employed to construct a good predictive model.

The task of predicting DNA-binding proteins using machine learning, involves two important steps: *i*) extraction of relevant features; and *ii*) selection of an appropriate classification algorithm. Depending on the feature extraction mechanism, the existing predictive methods can be classified into two different categories: *i*) extraction of relevant features from the structure of the protein [1, 31-33]; and *ii*) extraction of relevant features from the sequence of amino acids [10, 34-36]. According to Xu *et al.* [37], the accuracy of structure-based prediction methods for DNA-binding proteins is usually higher but, those cannot be used in high throughput annotation because those require a high-resolution 3D structure of the protein sequence. Thus, to date, various computational methods have been proposed for identifying DNA-binding proteins directly from their amino acid sequences. These methods independently explore four different categories of protein sequence features and four different kinds of sequence encoding methods [37-41]. Specifically, the four different categories of features include: (a) composition information; (b) structural and functional information; (c) physicochemical properties; and (d) evolutionary information. The four different kinds of encoding methods are: (a) overall composition-transition-distribution called OCTD (global method); (b) autocross-covariance (ACC) transformation (nonlocal method); (c) split amino acid (SAA) transformation (local method); and *(d)* PSSM distance transformation called PSSM-DT. These methods have been studied in detail in the relevant research work [37, 42, 43]. However, most of the proposed methods are limited in their ability to explain how protein-DNA interactions occur. Thus, it is essential to identify new features, effective encoding techniques and advanced machine learning techniques that can help further improve our understanding of DNA-protein interactions with higher accuracy.

This study explores, different features, encoding techniques and machine learning approaches, to further improve the prediction accuracy and understand the binding mechanism of DNA-protein interaction. The proposed method, StackDPPred, utilizes two different types of features: PSSM profile and residue wise contact energy profile. It uses four different types of feature transformation techniques: PSSM-DT, residue probing transformation [44], evolutionary distance transformation [38], residue wise contact energy matrix transformation [38], and a machine learning ensembles, known as stacking [45]. The development of StackDPPred offered a significant improvement in prediction accuracies based on the benchmark and independent test data when compared to the state-of-the-art predictors.

2.2 Methods

This section describes the procedure for benchmark and independent test data preparation, feature extraction and encoding, performance evaluation metrics and machine learning framework development.

2.2.1 Dataset

The benchmark dataset [37] that contains 525 DNA-binding protein sequences and 550 non DNA-binding protein sequences is used. In this implementation, only 518 DNA-binding proteins and 545 non DNA-binding proteins are used as the PSSM profile of 7 DNA-binding proteins and 5 non DNA-binding proteins using PSI-BLAST [46] could not be obtained.

An independent test dataset PDB186 [40] is used to compare the performance of StackDPPred with the state-of-the-art approaches. This dataset consists of 93 DNA-binding proteins and 93 non-DNA-binding proteins selected from the PDB to validate the quality of predictions.

2.2.2 Feature Extraction

To create an effective machine learning method to predict DNA-binding proteins from sequence alone, various features derived from the Position Specific Scoring Matrix (PSSM) and the Reside Wise Contact Energy Matrix (RCEM) are used, which are described next.

2.2.2.1 Position Specific Scoring Matrix (PSSM) feature

The PSSM captures the conservation pattern in multiple alignments and stores it as a matrix of scores for each position in the alignment. High scores represent more conserved positions and scores close to zero or negative, represent weakly conserved positions. Thus, PSSM captures the evolutionary information in proteins. Evolutionary information is one of the most important kinds of information for protein functionality annotation in biological analysis and is widely used in many studies [1, 22, 47-50]. For this study, the PSSM profile of every protein sequence is obtained by executing three iteration of PSI-BLAST against NCBI's non-redundant database [46]. The PSSM profile is a matrix of L*20 dimensions, which can be represented as follows:

$$PSSM = \begin{bmatrix} P_{1,1} & P_{1,2} & \cdots & P_{1,20} \\ P_{2,1} & P_{2,2} & \cdots & P_{2,20} \\ \vdots & \vdots & \vdots & \vdots \\ P_{L,1} & P_{L,2} & \dots & P_{L,20} \end{bmatrix} \tag{1}$$

where, *L* is the length of the protein and $P_{i,j}$ represents the occurrence probability of amino acid *i* at position *j* of the protein sequence, the rows represent the position of amino acid in the sequence and the columns represent the 20 standard amino acid types. The large positive scores indicate conserved positions, which in turn implies critical functional residues that are required to perform various intermolecular interactions.

2.2.2.2 Residue Wise Contact Energy Matrix (RCEM) feature

Structural stability of proteins is the result of a large number of inter and intra-residual interactions. The energy contribution of these interactions can be approximated by the energy functions extracted from known structures [1, 49, 51-55].

Table 1: RCEM table used in the proposed experiment.

	A	C	D	E	F	G	H	I	K	L	M	N	P	Q	R	S	T	V	W	Y
A	-1.65	-2.83	1.16	1.8	-3.73	-0.41	1.9	-3.69	0.49	-3.01	-2.08	0.66	1.54	1.2	0.98	-0.08	0.46	-2.31	0.32	-4.62
C	-2.83	-39.5	-0.82	-0.53	-3.07	-2.96	-4.98	0.34	-1.38	-2.15	1.43	-4.18	-2.13	-2.91	-0.41	-2.33	-1.84	-0.16	4.26	-4 .46
D	1.16	-0.82	0.84	1.97	-0.92	0.88	-1.07	0.68	-1.93	0.23	0.61	0.32	3.31	2.67	-2.02	0.91	-0.65	0.94	-0.71	0. 90
E	1.8	-0.53	1.97	1.45	0.94	1.31	0.61	1.3	-2.51	1.14	2.53	0.2	1.44	0.1	-3.13	0.81	1.54	0.12	-1.07	1. 29
F	-3.73	-3.07	-0.92	0.94	-11.2	0.35	-3.57	-5.88	-0.82	-8.59	-5.34	0.73	0.32	0.77	-0.4	-2.22	0.11	-7.05	-7.09	-8.80
G	-0.41	-2.96	0.88	1.31	0.35	-0.2	1.09	-0.65	-0.16	-0.55	-0.52	-0.32	2.25	1.11	0.84	0.71	0.59	-0.38	1.69	-1.90
H	1.9	-4.98	-1.07	0.61	-3.57	1.09	1.97	-0.71	2.89	-0.86	-0.75	1.84	0.35	2.64	2.05	0.82	-0.01	0.27	-7.58	-3.20
I	-3.69	0.34	0.68	1.3	-5.88	-0.65	-0.71	-6.74	-0.01	-9.01	-3.62	-0.07	0.12	-0.18	0.19	-0.15	0.63	-6.54	-3.78	-5.26
K	0.49	-1.38	-1.93	-2.51	-0.82	-0.16	2.89	-0.01	1.24	0.49	1.61	1.12	0.51	0.43	2.34	0.19	-1.11	0.19	0.02	-1.19
L	-3.01	-2.15	0.23	1.14	-8.59	-0.55	-0.86	-9.01	0.49	-6.37	-2.88	0.97	1.81	-0.58	-0.6	-0.41	0.72	-5.43	-8.31	-4.90
M	-2.08	1.43	0.61	2.53	-5.34	-0.52	-0.75	-3.62	1.61	-2.88	-6.49	0.21	0.75	1.9	2.09	1.39	0.63	-2.59	-6.88	-9.73
N	0.66	-4.18	0.32	0.2	0.73	-0.32	1.84	-0.07	1.12	0.97	0.21	0.61	1.15	1.28	1.08	0.29	0.46	0.93	-0.74	0.93
P	1.54	-2.13	3.31	1.44	0.32	2.25	0.35	0.12	0.51	1.81	0.75	1.15	-0.42	2.97	1.06	1.12	1.65	0.38	-2.06	-2.09
Q	1.2	-2.91	2.67	0.1	0.77	1.11	2.64	-0.18	0.43	-0.58	1.9	1.28	2.97	-1.54	0.91	0.85	-0.07	-1.91	-0.76	0.01
R	0.98	-0.41	-2.02	-3.13	-0.4	0.84	2.05	0.19	2.34	-0.6	2.09	1.08	1.06	0.91	0.21	0.95	0.98	0.08	-5.89	0.36
S	-0.08	-2.33	0.91	0.81	-2.22	0.71	0.82	-0.15	0.19	-0.41	1.39	0.29	1.12	0.85	0.95	-0.48	-0.06	0.13	-3.03	-0.82
T	0.46	-1.84	-0.65	1.54	0.11	0.59	-0.01	0.63	-1.11	0.72	0.63	0.46	1.65	-0.07	0.98	-0.06	-0.96	1.14	-0.65	-0.37
V	-2.31	-0.16	0.94	0.12	-7.05	-0.38	0.27	-6.54	0.19	-5.43	-2.59	0.93	0.38	-1.91	0.08	0.13	1.14	-4.82	-2.13	-3.59
W	0.32	4.26	-0.71	-1.07	-7.09	1.69	-7.58	-3.78	0.02	-8.31	-6.88	-0.74	-2.06	-0.76	-5.89	-3.03	-0.65	-2.13	-1.73	-12.3
Y	-4.62	-4.46	0.9	1.29	-8.8	-1.9	-3.2	-5.26	-1.19	-4.9	-9.73	0.93	-2.09	0.01	0.36	-0.82	-0.37	-3.59	-12.3	-2.68

However, the energy functions require that the 3D structure be known in order to compute the summation of such energies. Thus, they are not applicable for proteins whose structure is not known or for intrinsically disordered proteins (IDPs) [56] . Moreover, it was found that IDPs are very common in DNA-binding proteins, particularly in disordered tails [44, 57]. Thus, to inherently incorporate important information regarding the amino acid interactions and intrinsically disordered regions (IDRs), the predicted residue wise contact energies are employed. These contact energies are derived in [44], using 674 protein's primary sequences by the least square fitting with the contact energies derived from tertiary structure of 785 proteins. The RCEM is a $\mathbf{20} \times \mathbf{20}$ dimensional matrix whose rows and columns represent 20 standard amino acids. Table 1. shows the RCEM table [44] that is used in this work.

2.2.3 Feature Extraction from PSSM profile

This section describes various feature extraction techniques utilized to obtain a fixed dimensional feature vector from the PSSM profile to encode protein sequences.

2.2.3.1 PSSM-Distance Transformation (PSSM-DT) feature

Two forms of PSSM distance transformation techniques [37] are used to transform the PSSM information into fixed dimensional vectors. Given a PSSM (Equation (1)) of protein sequences, two distance transformation schemes were used: *i*) the PSSM distance transformation for pairs of same amino acids (PSSM-SDT); and *ii*) the PSSM distance transformation for pairs of different amino acids (PSSM-DDT), to extract fixed sized feature vectors.

PSSM-SDT features approximately measure the occurrence probabilities for the pairs of the same amino acids separated by a distance *d* along the sequence in a sequence, which can be calculated as:

$$PSSM\text{-}SDT(j,d) = \sum_{i=1}^{L-d} P_{i,j} * \ P_{i+d,j}/(L-d) \qquad (2)$$

where, *j* is one type of the amino acid, *L* is the length of the sequence, $P_{i,j}$ is the PSSM score of amino acid *j* at position *i*, and $P_{i+d,j}$ is the PSSM score of amino acid *j* at position $i+d$. Through this approach, 20**D* number of PSSM-SDT features were generated, where *D* is the maximum range of *d* (*d* = 1, 2…, *D*).

Similarly, PSSM-DDT calculates the occurrence probabilities for pairs of different amino acids separated by a distance of d along the sequence, which can be calculated as:

$$PSSM\text{-}DDT(i_1, i_2, d) = \sum_{j=1}^{L-d} P_{j,i_1} * \ P_{j+d,\ i_2} / (L-d) \tag{3}$$

where, i_1 and i_2 represent two different types of amino acids. The total number of features obtained by PSSM-DDT are *380*D*. Furthermore, *D* = 5, was found [37] to perform the best. As previously published benchmark dataset is used in this study, the same value of *D* found in previous study [37] is used in our experiment.

2.2.3.2 Residue Probing Transformation (RPT) feature

RPT, proposed by Jeong *et al.* [58], emphasizes domains with similar conservation rates by grouping domain families based on their conservation score in the PSSM profile. Here, each probe is a standard amino acid, and corresponds to a particular column in the PSSM profile. The rows in the PSSM profile are divided into 20 groups according to 20 different standard amino acids. Then, for each group, the sum of the PSSM values in every column are computed, leading to a feature vector of a $\mathbf{20 \times 20}$ dimensional matrix, called RPT matrix, represented as in Equation (4):

$$RPT = \begin{bmatrix} S_{1,1} & S_{1,2} & \cdots & S_{1,20} \\ S_{2,1} & S_{2,2} & \cdots & S_{2,20} \\ \vdots & \vdots & \vdots & \vdots \\ S_{20,1} & S_{20,2} & \cdots & S_{20,20} \end{bmatrix} \tag{4}$$

The RPT matrix (Equation (4)) is then transferred into a feature vector of 400 dimensions, as shown in Equation (5):

$$V = [f_{s_{1,1}}, f_{s_{1,2}}, \cdots, f_{s_{i,j}}, \cdots, f_{s_{20,20}}] \tag{5}$$

where, $f_{s_{i,j}}$ is calculated using Equation (6):

$$f_{s_{i,j}} = \frac{s_{i,j}}{L} \; (i, j = 1,2, \cdots, 20) \tag{6}$$

2.2.3.3 Evolutionary Distance Transformation (EDT) feature

The EDT extracts the information of the non-co-occurrence probability for two amino acids separated by a certain distance *d* in a protein from the PSSM profile [38]. Here, *d* is the distance between these two amino acids in a sequence $(\boldsymbol{d} = \mathbf{1}, \mathbf{2}, \ldots, \boldsymbol{L_{min}}\text{-}\mathbf{1})$, where $\boldsymbol{L_{min}}$ is the length of the shortest protein in the benchmark dataset. For example, $\boldsymbol{d} = \mathbf{1}$ implies that the two amino acids are consecutive; $\boldsymbol{d} = \mathbf{2}$ implies that there is one amino acid between the two, and so on till d = $\boldsymbol{L_{min}}\text{-}\mathbf{1}$. The EDT feature vector computed from the PSSM profile can be represented as (7):

$$P = [\partial_1, \partial_2, \cdots, \partial_\Omega] \tag{7}$$

where, Ω is an integer that represents the dimension of the vector whose value is 400. The non-co-occurrence probability of two amino acids separated by distance *d* (here, *d* ranges from 1 to *D*, where, *D* is the maximum value of *d*) can be computed using Equation (8):

$$f(A_x, A_y) = \sum_{d=1}^{D} \frac{1}{L-d} \sum_{i=1}^{L-d} (P_{i,x} - P_{i+d,y})^2 \tag{8}$$

where, $P_{i,x}$ and $P_{i+d,y}$ are the elements in the PSSM profile; A_x and A_y represent any of the 20 different amino acids in the protein. Finally, each element in feature vector *P* is obtained as given in Equation (9):

$$\{\partial_1 = f(A_1, A_2)\} \tag{9}$$

$$\{\partial_{400} = f(A_{20}, A_{20})\}$$

2.2.4 Feature extracted from RCEM

Depending on the type of amino acid, a 20-dimensional vector for each amino acid in a sequence is obtained from the rows of RCEM (see Table 1). Thus, for a protein sequence of length L, an L × 20-dimensional matrix E_m is obtained. Next, the matrix E_m is transformed into a feature vector of 20 dimensions by calculating the column-wise sum. If the element of matrix E_m is represented by $e_{i,j}$, where *i* represents the amino acid index in a sequence (rows) and *j* represents 20 standard amino acid types (columns), the column-wise sum of matrix E_m can be obtained using Equation (10):

$$f(A_j) = \sum_{i=1}^{L} e_{i,j} \ (j = 1,2,\cdots,20) \tag{10}$$

Then, the final feature vector, RCEM-Transformation ($RCEMT$) $= [E_1, E_2, \dots, E_{20}]$ is obtained by dividing each element in RCEMT by the total sum of the elements in the same vector, which can be represented as shown in Equation (11):

$$RCEMT(E_i) = \frac{E_i}{L} \tag{11}$$

2.2.5 Performance Evaluation

The performance of StackDPPred is measured by the jackknife validation approach. In jackknife validation, every sample is tested by the predictor trained with all the other samples in the benchmark set. Various performance evaluation measures listed in the Table 2 are employed to test the predictive capacity of the proposed method as well as to compare it with the state-of-the-art methods.

Table 2: Name and definition of the evaluation metric.

Name of Metric	**Definition**
True Positive (TP)	Correctly predicted DNA-binding proteins
True Negative (TN)	Correctly predicted non DNA-binding proteins
False Positive (FP)	Incorrectly predicted DNA-binding proteins
False Negative (FN)	Incorrectly predicted non DNA-binding proteins
Recall/Sensitivity/True Positive Rate (TPR/SN)	$\frac{TP}{TP + FN}$
Specificity/True Negative Rate (TNR/SP)	$\frac{TN}{TN + FP}$
Fall-out Rate/False Positive Rate (FPR)	$\frac{FP}{FP + TN}$
Miss Rate/False Negative Rate (FNR)	$\frac{FN}{FN + TP}$
Accuracy (ACC)	$\frac{TP + TN}{FP + FP + TN + FN}$
Balanced Accuracy (Bal_ACC)	$\frac{1}{2}(\frac{TP}{TP + FN} + \frac{TN}{TN + FP})$
Precision	$\frac{TP}{TP + FP}$
F1 score (Harmonic mean of precision and recall)	$\frac{2TP}{2TP + FP + FN}$
Mathews Correlation Coefficient (MCC)	$\frac{(TP * TN) - (FP * FN)}{\sqrt{(TP + FN) * (TP + FP) * (TN + FP) * (TN + FN)}}$
Strength (ST)	ST = (SN + SP)/2

Moreover, AUC and ROC performance measures are used to further evaluate the proposed method. AUC is the area under the receiver operating characteristics (ROC) curve and is commonly used to evaluate a predictor to see how well it separates two classes of information, i.e., in this case, DNA-binding versus non DNA-binding proteins.

2.2.6 Explored Base and Meta Classifiers

In order to find the base-classifiers to use in the first-stage and the meta-classifier to use in the second-stage of stacking framework, six different machine learning algorithms are explored and they are:

i) **Support Vector Machine (SVM):** SVM [39] with the radial basis function (RBF) kernel is used as one of the base-classifiers as well as a meta-classifier. SVM classifies by maximizing the separating hyperplane between two classes and penalizes the instances on the wrong side of the decision boundary using a cost parameter, *C*. The RBF kernel parameter, γ and the cost parameter, *C* are optimized to achieve the best jackknife validation accuracy using a grid search [59]. The best values of the parameters of the base-classifier SVM are found to be $C = 2^{3.50}$ and $\gamma = 2^{-11.25}$. Likewise, the best values of the parameters of the SVM, used as meta-classifier, are $C = 2^{11}$ and $\gamma = 2^{-16.25}$.

ii) **Logistic Regression (LogReg):** LogReg [60-62] with *L2* regularization is used as one of the base-classifiers. LogReg measures the relationship between the dependent variable, which is categorical (in our case: a protein being DNA-binding or not), and one or more independent variables by generating an estimation probability using logistic regression. The parameter, *C* which controls the regularization strength is optimized to achieve the best jackknife validation accuracy using grid search [59]. In my implementation, I found $C = 0.1$ results in the best accuracy.

iii) **Extra Trees (ET) Classifier:** Extremely randomized tree or ET [63] which is one of the ensemble methods is explored as a base-learner here. ET fits a number of randomized decision trees from the original learning sample and uses averaging to improve the predictive accuracy and control over-fitting. The ET model is constructed with 1,000 trees and the quality of a split is measured by the Gini impurity index.

iv) **Random Decision Forest (RDF) Classifier:** RDF [64] is used as one of the methods for base-classifiers. It operates by constructing a multitude of decision trees on various sub-samples of the dataset and outputs the mean prediction of the decision trees to improve the predictive accuracy and control over-fitting. In my implementation of the RDF ensemble learner, I have used bootstrap samples to construct 1,000 trees in the forest.

v) **K Nearest Neighbor (KNN) Classifier**: KNN [65] classifier is used as one of the methods for base-classifiers. The KNN operates by learning from the *K* number of training samples closest in distance to the target point in the feature space. The classification decision is produced based on the majority votes coming from the neighbors. In this work, the value of *K* is set to 9 and all the neighbors are weighted uniformly.

vi) **Bagging (BAG) Classifier:** Bootstrap aggregation or BAG [66] is explored as one of the methods for base-classifiers in this study. The BAG method forms a class of algorithms which builds several instances of a classifier/estimator on random subsets of the original training set and then aggregates their individual predictions to form a final prediction. The BAG method is useful for reducing variance in the prediction. In this study, the bagging classifier is fit on multiple subsets of data with the repetitions using 1,000 decision trees, and the outputs are combined by weighted averaging.

2.2.7 Framework of StackDPPred

In this study, a stacking technique [45] is applied to develop the StackDPPred predictor for DNA-binding proteins. Stacking is an ensemble technique used to combine information from multiple predictive models to generate a new model. It provides a scheme for minimizing the generalization error rate of one or more predictive models and has been successfully applied in several machine learning tasks [67-71].

Stacking framework involves two-stages of learning. The classifiers of the first-stage and second-stage are called base-classifier and meta-classifier respectively. A pool of base-classifiers is employed in the first-stage. Then, using meta-classifier in the second-stage, the outputs of the base-classifiers are combined with the aim of reducing the generalization error. To enrich the meta-classifier with more information on the solution space, it is desirable to use classifiers that are different from each other based on their underlying operating principle as the base-classifiers.

In order to find the base-classifiers to use in the first-stage and the meta-classifier to use in the second-stage of stacking framework, six different machine learning algorithms: (a) Support Vector Machine (SVM), (b) Logistic Regression (LogReg), (c) Extra Trees (ET), (d) Random Decision Forest (RDF), (e) K Nearest Neighbor (KNN) and (f) Bagging (BAG) are explored.

All of the above mentioned classifiers are built and tuned using scikit-learn [72]. To select a pool of algorithms to be used as the base-classifiers for the stacked model (SM), three different combinations of base-classifier are evaluated. The three different combinations of base-classifiers formed and tested in this study are:

i) **SM1:** includes SVM, LogReg, KNN and RDF,

ii) **SM2:** includes SVM, LogReg, KNN and ET, and

iii) **SM3:** includes SVM, LogReg, KNN and BAG.

As RDF, ET and BAG are tree based methods, they are individually combined with the other three methods, SVM, LogReg and KNN, whose underlying principles are different from each other. For all of the above combinations SM1, SM2 and SM3, the meta-level classifier is SVM. The jackknife validation of the above three combinations show that the SM1 when combined with SVM provides the best accuracy. Thus, four classifiers SVM, LogReg, KNN and RDF are employed as base-classifiers in the StackDPPred framework. The output probabilities (binding probability *p* and non-binding probability (1-*p*)) generated by the four base-classifiers are combined with the original 2,820 features and used as input features to train a new SVM classifier (meta-classifier) which is then used to obtain a robust StackDPPred classifier. The framework of StackDPPred is shown in Figure 1.

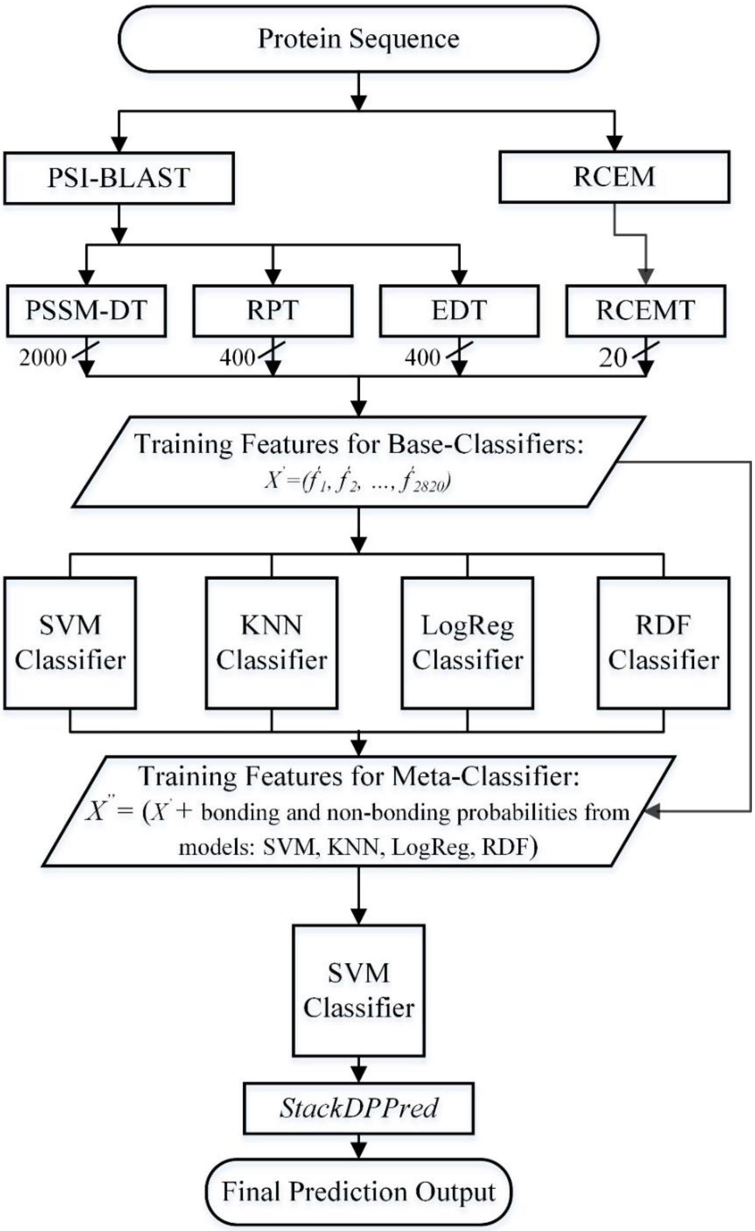
Protein Sequence
PSI-BLAST
RCEM
PSSM-DT
RPT
EDT
RCEMT
2000
400
400
20
Training Features for Base-Classifiers:
$X' = (f'_1, f'_2, \ldots, f'_{2820})$
SVM Classifier
KNN Classifier
LogReg Classifier
RDF Classifier
Training Features for Meta-Classifier:
$X'' = (X' +$ bonding and non-bonding probabilities from models: SVM, KNN, LogReg, RDF)
SVM Classifier
StackDPPred
Final Prediction Output

Fig. 1. Overview of the StackDPPred prediction framework.

2.3 Results

In this section, first the results of the potential base-classifiers for the development of StackDPPred is presented. Then, the performance of StackDPPred on the benchmark dataset and the independent test dataset is reported.

2.3.1 Selection of Base-classifiers

To select the best combination of classifiers to be used as the base-classifiers, first the performance of six different machine learning methods, SVM, LogReg, KNN, RDF, BAG and ET, on the benchmark dataset are analyzed. The individual predictive ability of each of the classifiers is obtained using the jackknife validation approach and is shown in Table 3.

Table 3 highlights that the optimized SVM with RBF-kernel gives an outstanding jackknife validation accuracy compared to other methods in this application. The SVM with RBF-kernel gives the best recall (completeness of the classifier in predicting DNA-binding proteins), specificity (completeness of the classifier in predicting non DNA-binding proteins), fall out rate (measures misclassification of non DNA-binding proteins as binding proteins), miss rate (measures misclassification of DNA-binding proteins as non DNA-binding proteins), balanced accuracy (measures overall completeness of the predictor), accuracy (measures the correctness of the predictor), precision (measures the exactness of the predictor), F1 score (measures overall correctness of the predictor) and MCC (another balanced measure of correctness of the predictor). Similarly, based on all of the performance measures provided in Table 3, LogReg provides the second highest performance. As the learning principle of SVM and LogReg are different from each other and they are the two highest performing methods, they are selected as two of the base-classifier initially. Next, out of the remaining four classifiers KNN, RDF, ET and BAG, the KNN is selected, which operates by learning from the K number of training samples closest in distance to the target point as the third base-classifier. Finally, one out of the three remaining ensemble based classifiers, RDF, ET and BAG, is selected as the fourth base-classifier and formulated 3 models, namely SM1, SM2, and SM3.

Table 3. Comparisons of various base learners on the benchmark dataset using jackknife cross-validation.

Metric/Method	LogReg	ET	KNN	SVM	BAG	RDF
Sensitivity	0.7851	0.6737	0.6602	**0.8030**	0.6988	0.6853
Specificity	0.7559	0.7522	0.7945	**0.8055**	0.7596	0.7761
Fall out rate	0.2440	0.2477	0.2055	**0.1945**	0.2403	0.2238
Miss Rate	0.2142	0.3262	0.3397	**0.1969**	0.3011	0.3146
Bal. Accuracy	0.7708	0.7130	0.7273	**0.8043**	0.7292	0.7307
Accuracy	0.7705	0.7385	0.7291	**0.8043**	0.7422	0.7440
Precision	0.7537	0.7210	0.7533	**0.7969**	0.7342	0.7442
F1 score	0.7693	0.6966	0.7037	**0.8000**	0.7161	0.7135
MCC	0.5415	0.4276	0.4594	**0.6084**	0.4595	0.4637

Table 4. Comparisons of stacked models with different set of base-classifiers through jackknife validation.

Method/ Metric	Sensitivity	Specificity	Fall out rate	Miss Rate	Bal. ACC	Accuracy	Precision	F1 score	MCC
SM1	0.911	0.888	0.111	0.088	0.899	0.899	0.898	0.899	0.799
SM2	0.899	0.884	0.115	0.100	0.892	0.891	0.880	0.890	0.783
SM3	0.901	0.882	0.117	0.098	0.892	0.891	0.879	0.890	0.783

Table 4 shows that the SM1, SM2 and SM3 stacked model categories provide similar performance. However, SM1, which includes SVM, LogReg, KNN and RDF as base-classifiers and another SVM as a meta-classifier provides the best performance. Thus, SM1 is selected as a final predictor.

2.3.2 Performance Comparison on Benchmark Dataset

This section compares the performance of StackDPPred with the state-of-the-art methods on the benchmark dataset. The quantities for all the evaluation metrics for the state-of-the-art methods, iDNA-Prot [17], DNABinder [37], DNA-Prot [73] and PSSM-DT[37] are obtained from Xu et al. [37].

Table 5. Comparisons of StackDPPred with other state-of-art methods on benchmark dataset through jackknife validation.

Method/Metric	ACC	Sensitivity	Specificity	MCC	AUC
PSSM-DT (imp. %)	0.7996 (12.50%)	0.8191 (11.24%)	0.7800 (13.9%)	0.6220 (28.5%)	0.8650 (11.56%)
iDNA-Prot (imp. %)	0.7540 (19.21%)	0.8381 (8.72%)	0.6473 (37.2%)	0.5000 (59.8%)	0.7610 (24.1%)
DNAbinder (imp. %)	0.7358 (22.26%)	0.6647 (37.08%)	0.8036 (10.5%)	0.4700 (70%)	0.8150 (15.95%)
DNA-Prot (imp. %)	0.7255 (24.0%)	0.8267 (10.22%)	0.5976 (48.6%)	0.4400 (81.5%)	0.7890 (19.8%)
StackDPPred (avg. imp. %)	**0.8996 (19.5%)**	**0.9112 (36.4%)**	**0.8880 (27.6%)**	**0.7990 (60.0%)**	**0.9449 (17.8%)**

Here, imp. and avg. stands for improvement and average respectively.

From Table 5, it can be observed that StackDPPred performs higher than all of the state-of-the-art methods. Specifically, StackDPPred provides 19.5%, 36.4%, 27.6%, 60.0% and 17.8% improvement on average over PSSM-DT, iDNA-Prot, DNAbinder and DNA-Prot methods based on ACC, sensitivity, specificity, MCC and AUC respectively.

Furthermore, Figure 2 presents the ROC curves generated by StackDPPred and PSSM-DT while the predictions are evaluated through jackknife validation on the benchmark dataset. The proposed predictor StackDPPred is only compared with PSSM-DT as it is the highest performing method among those listed in Table 5. The ROC curves show the TPR (sensitivity)/FPR (1-specificity) pairs at different classification thresholds. The curves highlight the strength of StackDPPred in achieving a high TPR of ≥ 85% (rate of correct prediction of DNA-binding proteins) at a very low FPR of ≤ 20%. Whereas, PSSM-DT provides TPR of ≥ 80% at a cost of higher FPR of ≥ 20%. Moreover, the AUC score given by StackDPPred is 9.30% higher than that of PSSM-DT when evaluated on benchmark dataset. These results show that StackDPPred is a promising predictor.

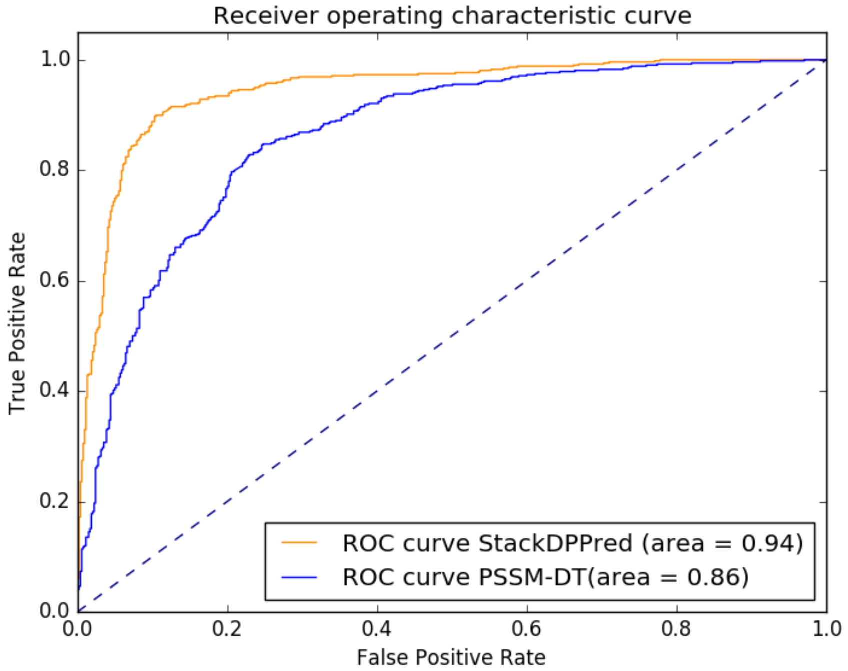

Fig. 2. Comparisons of ROC curves and AUC scores given by StackDPPred and PSSM-DT on benchmark dataset.

2.3.3 Performance Comparison using Independent Test Dataset

In this section, the performance of StackDPPred is further compared with the state-of-the-art methods on an independent test dataset, PDB186. The PDB186 dataset was recently constructed by Lou *et al* [40] to validate the quality of DNA-binding predictions. It consists of 93 DNA-binding proteins and an equal number of non-DNA-binding proteins. Some proteins on the benchmark dataset which share more than 25% sequence similarity to proteins in PDB186 dataset were removed using the program BLASTCLUST [74] to remove homologous bias. Then, StackDPPred was trained on the benchmark dataset and tested on the independent test dataset. Table 6 lists the predictive results of StackDPPred and the state-of-the-art methods including PSSM-DT [37], iDNA-Prot [17], DNA-Prot [73], DNAbinder [37], DNA-BIND [75], DNA-Threader [76] and DBPPred [40] on the PDB186 dataset.

Table 6. Comparisons of StackDPPred with the state-of-the-art methods on independent dataset, PDB186.

Methods	ACC	Sensitivity	Specificity	MCC	AUC
PSSM-DT (imp. %)	0.8000 (8.20%)	0.8709 (6.18%)	0.7283 (10.73%)	0.6470 (13.8%)	0.8740 (1.58%)
iDNA-Prot (imp. %)	0.6720 (28.8%)	0.6770 (36.6%)	0.6670 (20.9%)	0.3440 (114.1%)	0.8330 (6.58%)
DNA-Prot (imp. %)	0.6180 (40.06%)	0.6990 (32.3%)	0.5380 (49.9%)	0.2400 (206.8%)	0.7960 (11.5%)
DNAbinder (imp. %)	0.6080 (42.4%)	0.5700 (95.7%)	0.6450 (25.0%)	0.2160 (240.9%)	0.6070 (46.3%)
DNA-BIND (imp. %)	0.6770 (27.8%)	0.6670 (62.2%)	0.6880 (17.21%)	0.3550 (115.1%)	0.6940 (27.9%)
DBPPred (imp. %)	0.7690 (12.6%)	0.7960 (16.2%)	0.7420 (8.7%)	0.5380 (36.9%)	0.7910 (12.2%)
StackDPPred (avg. imp. %)	**0.8655 (26.7%)**	**0.9247 (41.5%)**	**0.8064 (22.1%)**	**0.7363 (121.3%)**	**0.8878 (11.7%)**

Here, imp. and avg. stands for improvement and average respectively.

Table 6 indicates that based on the independent test dataset the StackDPPred outperforms PSSM-DT, iDNA-Prot, DNA-Prot, DNAbinder, DNA-BIND and DNAPred methods on average by 26.7%, 41.5%, 22.1%, 121.3% and 11.7% based on ACC, sensitivity, specificity, MCC and AUC respectively.

Moreover, Figure 3 presents the ROC curves generated by StackDPPred and PSSM-DT while the predictions are evaluated on the independent test dataset. As PSSM-DT is the highest performing among the existing methods, the performance of StackDPPred is directly compared with PSSM-DT. The curves in Figure 2 highlight the strength of StackDPPred in achieving a high TPR of ≥ 85% (rate of correct prediction of DNA-binding proteins) at a very low FPR of around 20%. Whereas, PSSM-DT provides TPR of ≥ 80% at a cost of higher FPR ≥ 20%. In addition, the AUC score given by StackDPPred is 2.30% higher than that of PSSM-DT when evaluated on an independent test dataset. This indicates that the proposed method, StackDPPred outperforms the existing methods and is a very promising predictor.

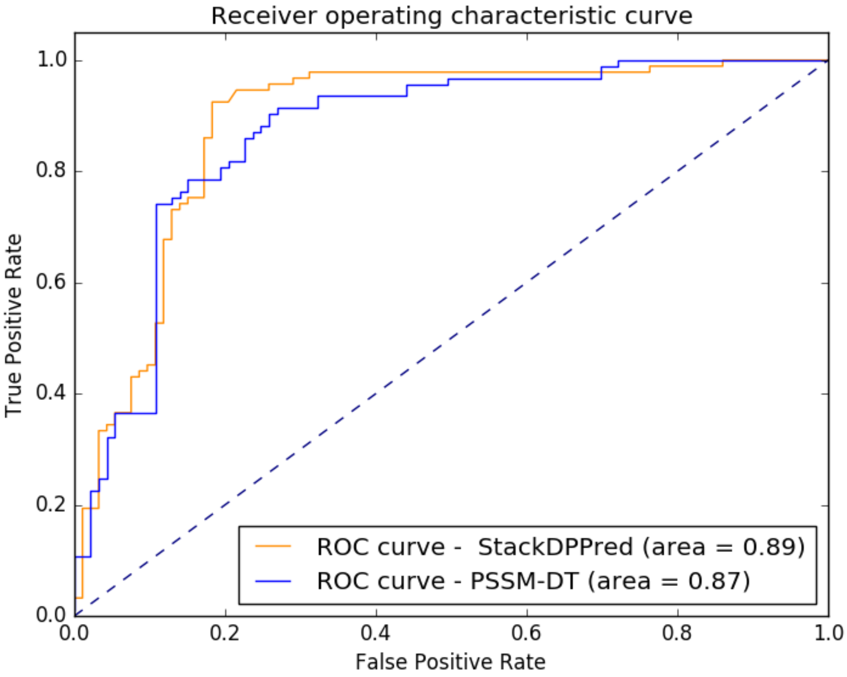

Fig. 3. Comparison of ROC curve and AUC scores given by StackDPPred and PSSM-DT on an independent test dataset.

2.4 Conclusions

This work presents a design and development of a stacking based machine learning technique, called StackDPPred, for the prediction of DNA-binding proteins given the protein sequence. With an aim to improve the prediction accuracy of DNA-binding proteins, different feature extractions and encoding techniques along with the advanced machine learning technique called stacking has been investigated and utilized. Important features such as evolutionary information feature and residue-wise contact energy are extracted from the PSSM and RCEM respectively. Next, these features are used to train the ensemble of predictors at the first-stage (base-layer). Then, the output of the predictors at the base-layer are combined using another SVM at the second-stage (meta-layer). As a result, the meta-learner SVM of the StackDPPred achieves an ACC of 89.96%, MCC of 0.7990 and AUC of 94.50% on a benchmark dataset, whereas the base-learner SVM achieves an ACC of 80.43% and MCC of 0.60849. This achievement, allows us to conclude that stacking technique helps reduce the generalization error and thus can improve accuracy significantly. As for the commonly used independent test dataset PDB186, StackDPPred attains an ACC of 86.56%, MCC of 0.7363 and AUC of 88.78%. This study reveals that the proposed method, StackDPPred, outperforms existing methods based on both benchmark validation and independent test datasets. These promising results indicate that StackDPPred can be effectively used for large-scale annotation of proteins based on DNA-binding affinity, given only the sequence information.

Chapter 3

Prediction of DNA-binding residues from sequences

3.1 Introduction

DNA-binding proteins play an important role in normal life cycle of an organism including DNA replication, transcription, repair, packaging and gene expression [77]. Various studies have reported that almost 2-3% of prokaryotic genome and 6-7% of eukaryotic genome encode DNA-binding proteins [4, 78]. Literature shows that the interactions between proteins and DNA occur through intermediate contacts [79]. Moreover, Bullock and Fersht [80] have shown that mutations of DNA-binding residues, such as those on the tumor repressor protein P53, may predispose individuals to cancer. It can thus be seen that the identification of DNA-binding residues is not only important for biological research (function annotation) but also for understanding pathogenesis of many diseases. Traditionally, protein-DNA interaction sites have been studied through biochemical experimental techniques such as nuclear magnetic resonance (NMR) spectroscopy [81], electrophoretic mobility shift assays (EMSAs) [82, 83], X-ray crystallography [84], peptide nucleic acid (PNA)-assisted identification of RNA binding proteins (RBPs) (PAIR) [85], MicroChIP [86], Fast ChIP [87], and conventional chromatin immunoprecipitation (ChIP) [88]. However, these methods are often laborious, time-consuming and often, costly. With rapidly increasing quantity of protein sequences, it is important to devise an objective computational method for the prediction of DNA-binding sites.

Various methods have been proposed in literature for the identification of DNA-binding sites in proteins. The features used in these prediction methods can be categorized into three types: sequence features, structure features and evolutionary features. In the early days, structural and sequence based features were mostly used for prediction since evolutionary features were hard to compute due to the lack of computing power. Some of the classifiers developed in the early days include the Support Vector Machine (SVM) classifier developed by Ahmad et al. [79], which utilized only sequence features, such as the local amino acid composition and solvent accessible surface area. Similarly, Tsuchiya et al. [89] used only structural features, such as electrostatic potential on the surface and the shape of the molecular surface. Likewise, Bhardwaj et al. [90] used both sequence and structure information, such as solvent accessibility, local composition, net charge, and electrostatic potentials. The later SVM classifier proposed by Bhardwaj et al. [91] used structure features such as the net charge of a residue, occurrence in a cationic patch, and the average potential on a residue in addition to the features used in their previous work [90]. It has been reported through various studies that evolutionary features are important for the identification of DNA-binding proteins [92] [93] [94]. The accuracy of the methods without evolutionary features is generally lower and the classifier is often skewed due to imbalanced number of samples. Thus, the inclusion of evolutionary information for the prediction of DNA-binding residues can improve the accuracy.

The last decade has seen a tremendous growth in the computing power which has made computing evolutionary features, which are often time consuming, much easier. Specifically, evolutionary features are represented in the form of Position Specific Scoring Matrix (PSSM) which are usually calculated in the two ways: (a) Concatenation methods that do encoding of the residues by concatenating PSSM scores in a sliding window, and (b) Combination methods that do the encoding of the residues by combining PSSM scores with other physiochemical properties like hydrophobicity, torsion angles, molecular mass and other frequency profiles. For example, the classifier proposed by Wang et al. [94] combined the three physiochemical properties including hydrophobicity, side chain pKa value, molecular mass and frequency profile to calculate the relevant physiochemical features for the target and context residues, and used their mean and standard deviation to construct the feature space. Similarly, Ma et al. [95]

combined the PSSM score of the residue with four physiochemical properties like lone electron pairs, hydrophobicity, side chain pKa value and molecular mass in their work. Likewise, concatenation methods have been used alone or in combination with other structural and physiochemical features to encode the residues. The work of Ahmad and Sarai [79] concatenated the PSSM scores of residues within a sliding window around the target residue to construct feature vectors. Similar methods have been used by several other classifiers. For example, the predictor called SVM-PSSM, proposed by Ho et al. [96] used a combination of concatenation methods, as well as other sequence and structural features. Likewise, the SVM classifier proposed by Ofran et al. [77] integrated the concatenation features and sequence features with solvent accessibility and predicted secondary structure for improved accuracy.

In this study, various properties about the protein sequences have been studied such as amino acid composition type, PSSM values of amino acids, physicochemical properties, predicted structural properties, torsion angles and the disorder values. To get as much information about the target and the context residues as possible, a sliding window was used and the features, concatenated to achieve a superior predictive performance. Furthermore, a Support Vector Machine (SVM) based predictor was used as the machine learning method for the classification task.

3.2 Materials and Methods

In this section, various data sources, data processing methods, input feature generation methods, algorithm design and performance evaluation for the prediction of DNA-binding residues are discussed.

3.2.1 Dataset Preparation

To evaluate the predictive performance of the proposed method for DNA-binding residue prediction and to compare it with other existing state-of-the-art classifiers, a widely used benchmark dataset PDNA-224 [97] for DNA-binding residue prediction was used. All the proteins in both datasets have sequence similarity ≤ 25% to each other.

3.2.2 Evaluation Metrics

The performance of the proposed method was measured using 5-fold cross validation. Various performance evaluation measures listed in the Table 2 to test the predictive capacity of the proposed method as well as to compare it with the state-of-the-art methods

3.2.3 Input Features

To obtain the relevant information about the residues, features that included useful properties like sequence information, evolutionary information as well as the predicted structural information (Table 1) were chosen. Some studies in the literature have revealed that the information about correct folding of protein is encoded in amino acid sequence and the disorder contents [98] . Because the information about the binding affinity of proteins is encoded not only in the evolutionary information, but also in other structural and physiochemical properties [32, 95, 99], these features were combined with evolutionary features for better prediction.

First, the amino acid type (AA), which is indicated by one numerical value out of twenty and other seven physicochemical properties(PP) were used as features to predict the DNA-binding residues. In addition to that, this study used the Position Specific Scoring Matrix (PSSM) as input features which was generated by running three iterations of PSI-BLAST [46] against the NCBI's non-redundant database [100]. The PSSM values were then normalized to get mean of zero and standard deviation of one using a PSSM normalizing factor of 9. Similarly, the predicted sequence based secondary structure (SS) for helix, sheet and coil residues were computed using the program SPINE-X [101]. Likewise, the predicted solvent accessibility (ASA) [102] and predicted backbone dihedral torsion angle fluctuations (Φ and Ψ) [103] were used to gain the information about the structural properties of the amino acids.

Table 7. List of features used in the prediction of DNA-binding residues from sequences

Feature Category	Feature Count
Amino Acid (AA)	1
Physicochemical Properties	7
PSSM profile	20
Secondary Structure Content	3
Accessible Surface Area	1
Torsion Angle Fluctuation	2
Monogram	1
Bigram	20
Disorder	2
Residue Wise Contact Energy Potential	20
Total	77

Since the information about the binding residues is encoded in the overall structure of the protein, the features which contribute to the overall three dimensional structure level should be included in the feature vectors by using the monograms and bigrams [104] as features. Various studies have shown that the conserved evolutionary information obtained from the amino acid sequence can be transformed to the three dimensional structure level by computing monograms and bigrams [1, 105]. Monogram feature matrix (1 * 20) and bigram feature matrix (20 * 20) for each sequence was calculated from its PSSM values. One monogram value and twenty bigram values for each type of amino acid were included as feature vectors. Studies suggest that the functional sites of the proteins that contribute to binding with other substrates is encoded in the order and disorder probability of amino acids [1]. Furthermore, the information about the disorder and structural stability of the proteins was incorporated by including disorder values from Dispredict [1] and the Residue Wise Contact Energy potential [52]. All the features used in this study are shown in Table 7.

The literature shows that the native interactions and the contacts of the neighboring residues play an important role in determining protein structure and protein folding dynamics [106, 107] and also in determining the DNA-binding preference of the residues [18, 79]. So, to get the information about the context residues within the feature space of each residue, the proposed method uses a sliding window with the target residue at the center of the window. The values in each column of the neighboring residues were then concatenated and then used as features for the proposed predictor. After wise, the optimal value of the sliding window size was determined by running the 3-fold cross validation on a randomly chosen sample from the benchmark dataset. The value of window size was experimentally found to be 11. The features were then scaled within the range [-1, +1] before being used for the prediction.

3.2.4 Machine Learning Method

Support Vector Machine(SVM) was used in this study as the machine learning method to capture the information about the DNA-binding residues

i) **Support Vector Machine (SVM):** SVM [39] with the radial basis function (RBF) kernel is used in this study. SVM classifies by maximizing the separating hyperplane between two classes and penalizes the instances on the wrong side of the decision boundary using a cost parameter, *C*. The RBF kernel parameter, γ and the cost parameter, *C* are optimized to achieve the best jackknife validation accuracy using a grid search [59].

3.3 Test Procedures and Results

3.3.1 Performance of 3-fold cross validation on the benchmark dataset for deciding the size of the sliding window

Because the benchmark dataset is so large and takes too long to run, only 10% of the sample was used to test for the sliding window.

Table 8. Performance of various window sizes on the benchmark dataset using the default SVM for DNA-binding residue prediction

Window / Metric	TPR	TNR	Bal ACC	ACC	FPR	FNR	Precision	F1	MCC
3	0.56414	0.83524	0.69969	0.81770	0.16476	0.43586	0.19150	0.28594	0.25013
5	0.64915	0.85203	0.75059	0.83890	0.14797	0.35085	0.23282	0.34272	0.32063
7	0.68006	0.85438	0.76722	0.84310	0.14562	0.31994	0.24417	0.35933	0.34205
9	0.70325	0.85630	0.77977	0.84640	0.14370	0.29675	0.25292	0.37204	0.35836
11	**0.72798**	**0.84668**	**0.78733**	**0.83900**	**0.15332**	**0.27202**	**0.24724**	**0.36912**	**0.35998**
13	0.72025	0.83941	0.77983	0.83170	0.16059	0.27975	0.23679	0.35641	0.34628
15	0.73261	0.81792	0.77527	0.81240	0.18208	0.26739	0.21773	0.33569	0.32817
17	0.74652	0.82016	0.78334	0.81540	0.17984	0.25348	0.22309	0.34353	0.33847
19	0.75270	0.81514	0.78392	0.81110	0.18486	0.24730	0.21977	0.34020	0.33633
21	0.75580	0.80905	0.78242	0.80560	0.19095	0.24420	0.21495	0.33470	0.33145
23	0.74498	0.80030	0.77455	0.80030	0.19587	0.25502	0.20830	0.32557	0.32030
25	0.73261	0.80274	0.76767	0.79820	0.19726	0.26739	0.20440	0.31962	0.31204
27	0.73725	0.79001	0.76363	0.78660	0.20999	0.26275	0.19541	0.30894	0.30195
29	0.75580	0.79215	0.77397	0.78980	0.20785	0.24420	0.20099	0.31753	0.31415
31	0.75116	0.78830	0.76973	0.78590	0.21170	0.24884	0.19708	0.31224	0.30788
33	0.75580	0.78274	0.76927	0.78100	0.21726	0.24420	0.19397	0.30871	0.30510
35	0.76507	0.77066	0.76787	0.77030	0.22934	0.23494	0.18750	0.30119	0.29897
37	0.76507	0.76211	0.76359	0.76230	0.23789	0.23493	0.18199	0.29403	0.29143
39	0.75425	0.76136	0.75781	0.76090	0.23864	0.24575	0.17941	0.28987	0.28504

The training data was shuffled at the residue level and the samples were taken in random to get a general representative sample of the whole dataset. The performance of the proposed method was then evaluated using a default SVM with different size of sliding window to determine the optimal window size. The values obtained are displayed in the Table 8.

It can be seen from Table 8 that the optimal performance is found at the window size of 11 which has Sensitivity of 0.728, Specificity of 0.847, Balanced Accuracy of 0.787 and Overall Accuracy of 0.839. The best Sensitivity is obtained at window size 34 and 37 which is 0.765 and best Specificity is obtained at the window size of 9 which is 0.856. However, the window size of 11 was used since it provides the best balanced accuracy which is an important metric for measuring the predictive performance of various machine learning methods in an imbalanced dataset. The classifier is then optimized in the next section which gives better performance than the un-optimized Support Vector Machine (SVM) classifier.

3.3.2 Performance of 5-fold cross validation on the benchmark dataset

Note that the preliminary extensive analysis of the performance of various window sizes is done using default parameters for SVM. For each run of the algorithm with the total number of residues ($Residue_{total}$) in a dataset, the feature matrix of dimension $Residue_{total} \times 77$ was used. To trade off between the performance and time complexity of parameter selection and model generation, the optimum parameters for the SVM were selected using 10% of the benchmark dataset. SVM with the penalty for class weights was used to find the optimum values of C and gamma and generate the model. The main motivation behind using the class weight penalty parameter is to balance the effect of highly skewed dataset which contains a high number of non DNA-binding residues and a very low number of DNA-binding residues. The best values of C and gamma is found to be 2^{-1} and 2^{-5} using gridsearch on the benchmark dataset. The improvement of performance with the optimized parameters over the non-optimized parameters was significant. The optimized model obtained an Accuracy of 0.9331 which is better than any other method proposed in the dataset. Finally, the optimal values of C and gamma was used to train the SVM on the whole dataset using 5 fold cross validation. Table 9 states the performance of the proposed method with the other state-of-the-art methods on the benchmark dataset.

Table 9. Performance of the SVM based method with other state-of-the-art methods on the benchmark dataset using 5-fold cross-validation for DNA-binding residue prediction

Methods	ACC	MCC	TPR	TNR	ST
Ma et al. (% imp)	0.7666 (21.74)	0.27 (96.5)	0.6895 (45.03)	0.7725 (12.18)	0.7310 (27.68%)
Li et al (% imp)	0.7865 (18.67)	0.29 (83.0)	0.6948 (43.93)	0.7934 (9.22)	0.7441 (25.42%)
EL_PSSM-RT (% imp)	0.7809 (19.51)	0.34 (56.08)	0.7958 (25.66)	0.7798 (11.13)	0.7878 (18.47%)
Proposed method (avg.% imp)	**0.93331 (19.97)**	**0.53067 (78.53)**	**1.0000 (38.20)**	**0.86662 (10.84)**	**0.93331 (23.86)**

Here, imp. stands for improvement and avg stands for average.

Table 9 shows that the proposed method significantly outperforms Ma et al [95], Li et al [97] and EL_PSSM-RT[108] on the benchmark dataset. Specifically, the proposed method outperforms other state-of-the-art approaches by 19.97% for ACC, 78.53% for MCC, 38.20% for Sensitivity, 10.84% for specificity and 23.86% for Strength. Thus, the proposed method can be efficiently used for prediction of DNA-binding residues.

3.4 Future Work

Since the thesis timeline didn’t allow me to perform the test on the independent dataset, the future works will measure the performance of our proposed method on the independent test dataset PDNA-62 [79]. The future work on the identification of DNA-binding residues will specifically focus on the complex stacking based architectures and several approaches to balance the dataset so as to improve the prediction accuracy. The methods will also be tested on several other datasets so as to establish the consistency in performance.

3.5 Conclusions

It has been observed through experimental analyses that DNA-protein interactions occur due to the immediate contacts between the DNA residues and the protein residues. Thus, the identification of the residues involved in the contacts is very important in order to understand the mechanism of DNA-protein interaction. In this study, the importance of various sequence-based, evolutionary, physicochemical, torsion angles and predicted structure based features were studied and the information about the context residues was included to design the predictor. It was found that the proposed approach outperforms all the other state-of-the-art methods on the benchmark dataset. The future work on the DNA-binding residues prediction problem will focus on using more benchmark and independent datasets and measure their performance to demonstrate the superiority of the proposed approach for identification of DNA-binding residues.

Chapter 4

Conclusions and Recommendations

In this work, I strived for the methodical discovery and characterization of new biological properties of proteomic data. I performed computational modeling of several structural and interaction properties of proteins to better understand their roles in biological processes through machine learning approaches. The comprehensive research objective addressed the applications in three disciplines:

1) **Bioinformatics**, which includes the development and implementation of various tools for bioinformatics applications using novel algorithms which will help in efficient access and management of different types of biological information.
2) **Computational Biology**, which includes the analysis and interpretation of various types of data including nucleotide and amino acid sequences, protein domains and their structures to perform an efficient analysis of biological features.
3) **Machine Learning**, which involves the development of novel learning algorithms to perform an advanced analysis of the data and get as much information from the feature space as possible.

In this chapter, I first present a quick summary of the contributions and then some directions for future research and finally conclude by some concluding remarks.

4.1 Summary

In this section, the contributions of this thesis are summarized.

StackDPPred: I have developed a stacked predictor for the prediction of DNA-binding proteins from sequence information only. In this research, I performed large scale proteomic data collection, purification and analysis from multiple sources such as UniProt [11] and NCBI non-redundant database. To develop the predictor, various machine learning methods like Support Vector Machines, Logistic Regression, k Nearest Neighbors, Random Forest were used. I carried out the tuning of parameters of all these methods through grid search and used the optimal values to develop our predictor.

SVM based predictor of DNA-binding residues: In this section, a new SVM based predictor has been developed for the prediction of DNA-binding residues using only the sequence information. For this task, various novel features like monogram, bigrams, predicted structural features and residue-wise contact energy potential have been utilized to develop the predictor. The predictor outperforms all the other state-of-the-art approaches.

4.2 Future Scopes

This section discusses the future scope of the research that has been conducted under this thesis. The possible future directions (but not limited to) are the following:

Section 2.3 of Chapter 2, discusses the possibility of combining various machine learning methods for efficient prediction of DNA-binding proteins using sequence information only. It has been observed that when machine learning methods are combined according to their operating principle, superior performance is obtained over using individual methods. Moreover, the StackDPPred only uses the features from the PSSM matrix and the RCEM matrix to obtain the proposed accuracy. The accuracy of the predictor can be further improved by employing important novel structural and sequence-based features as well as by including more training samples. Furthermore, it would be interesting to see the performance of our method with an extended pool of base learners.

Chapter 2 discusses the performance of the Support Vector Machine (SVM) based architecture for the prediction of DNA-binding residues using only sequence information. It is observed that with the inclusion of relevant features and the use of better machine learning methods can significantly improve the prediction accuracy. The methods used in this study will further be incorporated with other complex stacking based learners to significantly improve the predictive capability to make the predictors as near-accurate as possible.

Moreover, it would be interesting to see the performance of the novel stacking methods on other datasets, including but not limited to other prediction problems in bioinformatics.

4.3 Conclusions

The development of various computational methods for large-scale fast annotation and analysis of biological data to study the structure and function of proteins from sequence information, such as the ones developed in this thesis, can help guide and later, outgrow the use of experimental techniques for these tasks. The methods used in this study could be further extended to the analysis and the study of other kinds of proteins serving other functions. Although the methods presented in this thesis don't provide near-optimal performance, with the availability of more data and with the use of advanced methods, these tools could be used as near-accurate optimizers for the prediction of DNA-binding proteins and their binding sites.

To conclude, to pursue a predictive understanding of the information about the binding affinity of the proteins, development of computational frameworks based on the solid mathematical foundations and algorithms as well as the statistical evaluation is important. Fast and efficient annotation of the functional properties of the proteins can help us keep up with the rapid pace of biological research and furthermore, will contribute to the applications in other sciences and engineering domains involving predictive understanding and reasoning. All the tools, datasets and code for the tools developed under this thesis are publicly available as open source. I hope that these contributions, particularly StackDPPred and the SVM based DNA-binding residue predictor will serve as useful tools for advancing the computing as well as biological sciences, particularly in proteomics research and applications using machine learning.

References

1. Iqbal, S. and M.T. Hoque, *DisPredict: A Predictor of Disordered Protein Using Optimized RBF Kernel.* Plos One, 2015. **10**.
2. Luscombe, N.M., et al., *An overview of the structures of protein-DNA complexes.* Genome Biology, 2000.
3. Lin, C., et al., *Hierarchical Classification of Protein Folds Using a Novel Ensemble Classifier.* Plos One, 2013.
4. Walter, M.C., et al., *PEDANT covers all complete RefSeq genomes.* Neucleic Acids Res, 2008(37).
5. Harris, T.D., et al., *Single-molecule DNA sequencing of a viral genome.* Science, 2008. **320**(5872): p. 4.
6. Margulies, M., et al., *Genome sequencing in microfabricated high-density picolitre reactors.* nature, 2005. **437**.
7. Shendure, J., et al., *Accurate Multiplex Polony Sequencing of Bacterial Genome.* Science, 2005. **309**.
8. Wheeler, D.A., et al., *The complete genome of an individual by massively parallel DNA sequencing.* Nature. **452**.
9. Liolios, K., et al., *The Genomes On Line Database (GOLD) v. 2: a monitor of genome projects worldwide.* Nucleic Acids Research, 2006. **34**: p. D 332–D 334.
10. Zou, Q., et al., *Survey of MapReduce frame operation in bioinformatics.* Briefings in Bioinformatics
11. Wu, C.H., et al., *The Universal Protein Resource (UniProt): an expanding universe of protein information.* Nucleic Acids Research, 2006(34).
12. Gao, M. and J. Skolnick, *DBD-Hunter: a knowledge-based method for the prediction of DNA-protein interactions.* nucleic Acids Research, 2008. **36**(12): p. 3978-3992.
13. Shanahan, H.P., et al., *Identifying DNAbinding proteins using structural motifs and the electrostatic potential.* Nucleic Acids Research. **32**(16): p. 4732–4741.
14. Marcotte, E.M., et al., *Detecting Protein Function and Protein-Protein Interactions from Genome Sequences.* Science, 1999. **285**(5428).
15. Brown, J. and T. Akutsu, *Identification of novel DNA repair proteins via primary sequence, secondary structure, and homology.* BMC Bioinformatics, 2009. **10**(25).
16. Bhardwaj, N., et al., *Kernel-based machine learning protocol for predicting DNA-binding proteins.* Nucleic Acids Research. **33**(20): p. 6486-6493.
17. Huang, H.-L., et al., *Predicting and analyzing DNA-binding domains using a systematic approach to identifying a set of informative physicochemical and biochemical properties.* The Ninth Asia Pacific Bioinformatics Conference, 2011. **12**.
18. Xiong, Y., J. Liu, and D.-Q. Wei, *An accurate feature-based method for identifying DNA-binding residues on protein surfaces.* Proteins. **79**(2): p. 509-517.
19. Andrabi, M., A. Sarai, and S. Ahmad, *Prediction of mono- and dinucleotide-specific DNA-binding sites in proteins using neural networks.* BMC Structural Biology. **9**(30).
20. Stawiski, E.W., L.M. Gregore, and Y. Mandel-Gutfreund, *Annotating nucleic acid binding function based on protein structure.* Journal of Molecular Biology, 2003. **326**(4).
21. Ahmad, S. and A. Sarai, *Moment-based prediction of DNA-binding proteins.* Journal of Molecular Biology. **341**(1): p. 65-71.
22. Manish Kumar, M.M.G., , Gajendra PS Raghava, *Identification of DNA-binding proteins using support vector machines and evolutionary profiles.* BMC Bioinformatics, 2007. **9**(463).
23. Wei, L., et al., *Improved and Promising Identification of Human MicroRNAs by Incorporating a High-quality Negative Set.* IEEE/ACM Transactions on Computational Biology and Bioinformatics, 2013. **11**(1): p. 192 - 201.
24. Nimrod, G., et al., *iDBPs: a web server for the identification of DNA-binding proteins.* Bioinformatics. **25**(5).
25. Yan, C., et al., *Predicting DNA-binding sites of proteins from amino acid sequence.* BMC Bioinformatics, 2006. **7**.

26. Govindan, G. and A.S. Nair, *New Feature Vector for Apoptosis Protein Subcellular Localization Prediction*, in *International Conference on Advances in Computing and Communications*. 2011.
27. Qian, Z., Y.-D. Cai, and Y. Li, *A novel computational method to predict transcription factor DNA-binding preference.* Biochemical and Biophysical Research Communications. **348**(3): p. 1034-1037.
28. Nanni, L. and A. Lumini, *Combining ontologies and dipeptide composition for predicting DNA-binding proteins.* Amino Acids, 2015. **34**(4): p. 635–641.
29. Xia, J.-F., X.-M. Zhao, and D.-S. Huang, *Predicting protein-protein interactions from protein sequences using meta predictor.* Amino Acids. **39**(5): p. 1595-1599.
30. Zou, Q., et al., *BinMemPredict: a Web server and software for predicting membrane protein types.* Current Proteomics, 2013. **10**(1).
31. Tjong, H. and H.-X. Zhou, *DISPLAR: an accurate method for predicting DNAbinding sites on protein surfaces.* Nucleic Acids Research, 2007. **35**.
32. Liu, B., et al., *Using Amino Acid Physicochemical Distance Transformation for Fast Protein Remote Homology Detection.* Plos One, 2012. **7**.
33. Liu, B., et al., *A Discriminative Method for Protein Remote Homology Detection and Fold Recognition Combining Top-n-grams and Latent Semantic Analysis.* BMC Bioinformatics, 2008. **9**.
34. Feng, Z.-P. and C.-T. Zhang, *Prediction of membrane protein types based on the hydrophobic index of amino acids.* Journal of Protein Chemistry. **19**(4).
35. Wang, B., et al., *Predicting protein interaction sites from residue spatial sequence profile and evolution rate.* FEBS Letters, 2005. **580**(2): p. 380-384.
36. Moroni, E., M. Caselle, and F. Fogolari, *Identification of DNA-binding protein target sequences by physical effective energy functions: free energy analysis of lambda repressor-DNA complexes.* BMC Structural Biology, 2007. **7**(61).
37. Xu, R., et al., *Identifying DNA-binding proteins by combining support vector machine and PSSM distance transformation.* BMC Systems Biology, 2015. **9**.
38. Zhang, L., X. Zhao, and L. Kong, *Predict protein structural class for low-similarity sequences by evolutionary difference information into the general form of Chou's pseudo amino acid composition.* Journal of Theoretical Biology, 2014. **355**: p. 105-110.
39. Vapnik, V.N., *An Overview of Statistical Learning Theory.* IEEE TRANSACTIONS ON NEURAL NETWORKS, 1999. **10**(5).
40. Lou, W., et al., *Sequence Based Prediction of DNA-Binding Proteins Based on Hybrid Feature Selection Using Random Forest and Gaussian Naïve Bayes.* Plos One, 2014. **9**(1).
41. Hui-Lin, H., et al., *Predicting and Analyzing DNA-Binding Domains Using a Systematic Approach to Identifying a Set of Informative Physicochemical and Biochemical Properties.* BMC Bioinformatics, 2011. **12**.
42. Nanni, L., S. Brahnam, and A. Lumini, *High performance set of PseAAC and sequence based descriptors for protein classification.* Journal of Theoretical Biology. **266**(1): p. 1-10.
43. Zhang, Z., S. Kochhar, and M.G. Grigorov, *Descriptor-based protein remote homology identification.* Protein Science. **14**(2).
44. Dosztányi, Z., et al., *The Pairwise Energy Content Estimated from Amino Acid Composition Discriminates between Folded and Intrinsically Unstructured Proteins.* Journal of Molecular Biology, 2005. **347**(4): p. 827-839.
45. Wolpert, D.H., *Stacked generalization.* Neural Networks, 1992. **5**(2): p. 241-259.
46. Altschul, S.F., et al., *Gapped BLAST and PSI-BLAST: a new generation of protein database search programs.* nucleic Acids Research, 1997. **25**(17).
47. Biswas, A.K., N. Noman, and A.R. Sikder, *Machine learning approach to predict protein phosphorylation sites by incorporating evolutionary information.* BMC Bioinformatics. **11**.
48. Verma, R., G.C. Varshney, and G.P.S. Raghava, *Prediction of mitochondrial proteins of malaria parasite using split amino acid composition and PSSM profile.* Amino Acids, 2010. **39**(1).

49. Iqbal, S., A. Mishra, and M.T. Hoque, *Improved prediction of accessible surface area results in efficient energy function application.* Journal of Theoretical Biology, 2015. **380**.
50. Islam, M.N., et al., *A balanced secondary structure predictor.* Journal of Theoretical Biology, 2015. **389**.
51. Mishra, A. and M.T. Hoque, *Three-Dimensional Ideal Gas Reference State Based Energy Function.* BMC Bioinformatics, 2017. **12**(2): p. 171-180.
52. Mishra, A., S. Iqbal, and M.T. Hoque, *Discriminate protein decoys from native by using a scoring function based on ubiquitous Phi and Psi angles computed for all atom.* Journal of Theoretical Biology, 2016. **398**.
53. Zhou, H. and J. Skolnick, *GOAP: A Generalized Orientation-Dependent, All-Atom Statistical Potential for Protein Structure Prediction.* Biophysics Journal, 2011. **101**(8).
54. Iqbal, S. and M.T. Hoque, *Estimation of Position Specific Energy as a Feature of Protein Residues from Sequence alone for Structural Classification.* Plos One, 2016. **11**(9): p. 1-23.
55. Tarafder, S., et al., *RBSURFpred: Modeling protein accessible surface area in real and binary space using regularized and optimized regression.* Journal of Theoretical Biology, 2018. **441**: p. 44-57.
56. Babu, M.M., et al., *Intrinsically disordered proteins: regulation and disease.* Current Opinion on Structural Biology, 2016. **21**(3).
57. Vuzman, D., Y. Hoffman, and Y. Levy, *Modulating protein-DNA interactions by post-translational modifications at disordered regions.* Biocomputing. **2012**: p. 188-199.
58. Jeong, J.c., X. Lin, and X.-w. Chen, *On Position-Specific Scoring Matrix for Protein Function Prediction.* IEEE/ACM Transactions on Computational Biology and Bioinformatics, 2011. **308**(15).
59. Bergstra, J. and Y. Bengio, *Random Search for Hyper-Parameter Optimization.* Journal of Machine Learning Research, 2012. **13**.
60. Hastie, T., R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning*. 2 ed. Springer Series in Statics. 2009: Springer-Verlag New York.
61. Meer, P., et al., *Robust regression methods for computer vision: A review* International Journal of Computer Vision, 1991. **6**(1): p. 59-70.
62. Szilágyi, A. and J. Skolnick, *Efficient Prediction of Nucleic Acid Binding Function from Low-resolution Protein Structures.* Journal of Molecular Biology, 2006. **358**(3): p. 922-933.
63. Geurts, P., D. Ernst, and L. Wehenkel, *Extremely randomized trees.* Machine Learning, 2006. **63**(1): p. 3-42.
64. Ho, T.K., *Random decision forests*, in *Document Analysis and Recognition, 1995., Proceedings of the Third International Conference on*. 1995, IEEE: Montreal, Que., Canada. p. 278-282.
65. Altman, N.S., *An Introduction to Kernel and Nearest-Neighbor Nonparametric Regression.* The American Statistician, 1992. **46**: p. 175-185.
66. Breiman, L., *Bagging predictors.* Machine Learning, 1996. **24**(2): p. 123-140.
67. Frank, E., et al., *Data mining in bioinformatics using Weka.* Bioinformatics, 2004. **20**: p. 2479-2481.
68. Gorman, B. *A Kaggler's Guide to Model Stacking in Practice*. 2016 [cited 2018; Available from: http://blog.kaggle.com/2016/12/27/a-kagglers-guide-to-model-stacking-in-practice/.
69. Nagi, S. and D.K. Bhattacharyya, *Classification of microarray cancer data using ensemble approach.* Network Modeling Analysis in Health Informatics and Bioinformatics, 2013. **2**(3): p. 159-173.
70. Hu, Q., et al., *A Stacking-Based Approach to Identify Translated Upstream Open Reading Frames in Arabidopsis Thaliana*, in *International Symposium on Bioinformatics Research and Applications*. 2015, Bioinformatics Research and Applications. p. 138-149.
71. Verma, A. and S. Mehta, *A comparative study of ensemble learning methods for classification in bioinformatics*, in *Cloud Computing, Data Science & Engineering - Confluence, 2017 7th International Conference on*. 2017, IEEE: Noida, India.

72. Pedregosa, F., et al., *Scikit-learn: Machine Learning in Python.* Journal of Machine Learning Research, 2011. **12**: p. 2825-2830.
73. Kandaswamy, K.K., G. Pugalenthi, and P.N. Suganthan, *DNA-Prot: Identification of DNA Binding Proteins from Protein Sequence Information using Random Forest.* Journal of Biomolecular Structure and Dynamics, 2011. **26**(6).
74. Camacho, C., et al., *BLAST+: architecture and applications.* BMC Bioinformatics. **10**(421).
75. Szilágyi, A. and J. Skolnick, *Efficient prediction of nucleic acid binding function from low-resolution protein structures.* Journal of Molecular Biology. **358**(3): p. 922-933.
76. Gao, M. and J. Skolnick, *A threading-based method for the prediction of DNAbinding proteins with application to the human genome.* Plos One, 2009. **5**.
77. Ofran, Y., V. Mysore, and B. Rost, *Prediction of DNA-binding residues from sequence.* Oxford Bioinformatics. **23**(12): p. i347-i353.
78. Luscombe, N.M., et al., *An overview of the structures of protein-DNA complexes.* Genome Biology.
79. Ahmad, S., M.M. Gromiha, and A. Sarai, *Analysis and prediction of DNA-binding proteins and their binding residues based on composition, sequence and structural information.* Oxford Bioinformatics, 2004. **20**(4): p. 477-486.
80. Bullock, A.N. and A.R. Fersht, *Rescuing the function of mutant p53.* Nature Cancer Reviews, 2001: p. 68-76.
81. Ponting, C.P., et al., *SMART: Identification and annotation of domains from signalling and extracellular protein sequences.* Nucleic Acids Research, 1998. **27**(1).
82. Jones, S., et al., *Using structural motif templates to identify proteins with DNA binding function.* Nucleic Acids Research, 2003. **31**(11).
83. Jones, S., et al., *Protein-DNA interactions: a structural analysis.* Journal of Molecular Biology, 1999. **287**(5).
84. Orengo, C., et al., *CATH – a hierarchic classification of protein domain structures.* Structure, 1997. **5**(8): p. 1093-1109.
85. Olson, W.K., et al., *DNA sequence-dependent deformability deduced from protein–DNA crystal complexes.* Proceedings of the National Academy of Sciences of the United States of America, 1998. **95**(19).
86. Luscombe, N.M., R.A. Laskowski, and J.M. Thornton, *Amino acid–base interactions: a three-dimensional analysis of protein–DNA interactions at an atomic level.* Nucleic Acids Research, 2001. **29**(13).
87. Mandel-Gutfreund, Y. and H. Margalit, *Quantitative parameters for amino acid-base interaction: implications for prediction of protein-DNA binding sites.* Nucleic Acids Research, 1998. **26**(10).
88. H, K. and S. A., *Structure-based prediction of DNA target sites by regulatory proteins.* Proteins, 1999. **35**(1).
89. Tsuchiya, Y., K. Kinoshita, and H. Nakamura, *Structure-based prediction of DNA-binding sites on proteins Using the empirical preference of electrostatic potential and the shape of molecular surfaces.* Proteins, 2004.
90. Bhardwaj, N., et al., *Structure Based Prediction of Binding Residues on DNA-binding Proteins.* IEEE Xplore, 2006.
91. Bhardwaj, N. and H. Lu, *Residue-Level Prediction of DNA-Binding Sites and its Application on DNA-Binding Protein Predictions.* FEBS Letters, 2008. **581**(5).
92. Wang, L. and S.J. Brown, *BindN: a web-based tool for efficient prediction of DNA and RNA binding sites in amino acid sequences.* Nucleic Acids Research, 2006. **34**.
93. Ahmad, S. and A. Sarai, *PSSM-based prediction of DNA binding sites in proteins.* BMC Bioinformatics, 2005. **6**(33).
94. Wang, L., et al., *BindN+ for accurate prediction of DNA and RNA-binding residues from protein sequence features.* BMC Systems Biology, 2010. **4**.
95. Ma, X., et al., *A SVM-based approach for predicting DNA-binding residues in proteins from amino acid sequences.* IEEE Xplore, 2009.
96. Ho, S.-Y., et al., *Design of accurate predictors for DNA-binding sites in proteins using hybrid SVM–PSSM method.* Biosystems, 2007. **90**(1): p. 234-241.

97. Li, T., et al., *PreDNA: accurate prediction of DNA-binding sites in proteins by integrating sequence and geometric structure information.* Oxford Bioinformatics, 2013. **29**(6).

98. Jones, D.T. and J.J. Ward, *Prediction of disordered regions in proteins from position specific score matrices.* Proteins, 2003. **53**(6).

99. Huang, H.-L., et al., *Predicting and analyzing DNA-binding domains using a systematic approach to identifying a set of informative physicochemical and biochemical properties.* **12**.

100. Pruitt, K.D., et al., *NCBI Reference Sequences: current status, policy and new initiatives.* Nucleic Acids Research, 2009. **37**.

101. Faraggi, E., et al., *SPINE X: Improving protein secondary structure prediction by multi-step learning coupled with prediction of solvent accessible surface area and backbone torsion angles.* Journal of Computational Chemistry, 2013. **33**(3).

102. Faraggi, E., B. Xue, and Y. Zhou, *Improving the prediction accuracy of residue solvent accessibility and real-value backbone torsion angles of proteins by guided-learning through a two-layer neural network.* Proteins, 2009. **74**(4).

103. Zhang, T., E. Faraggi, and Y. Zhou, *Fluctuations of backbone torsion angles obtained from NMR-determined structures and their prediction.* Proteins, 2011. **78**(16).

104. Sharma, A., et al., *Evaluation of Sequence Features from Intrinsically Disordered Regions for the Estimation of Protein Function.* Plos One, 2014. **9**(2).

105. Sharma, A., et al., *A feature extraction technique using bi-gram probabilities of position specific scoring matrix for protein fold recognition.* Journal of Theoretical Biology, 2013. **320**.

106. Sun, Y. and D. Ming, *Energetic Frustrations in Protein Folding at Residue Resolution: A Homologous Simulation Study of Im9 Proteins.* Plos One, 2014. **9**(5).

107. Vendruscolo, M., et al., *Three key residues form a critical contact network in a protein folding transition state.* Nature International Journal of Science, 2001. **409**.

108. Zhou, J., et al., *EL_PSSM-RT: DNA-binding residue prediction by integrating ensemble learning with PSSM Relation Transformation.* BMC Bioinformatics, 2017. **18**(379).