Hindawi
BioMed Research International
Volume 2020, Article ID 7297631, 8 pages
https://doi.org/10.1155/2020/7297631

Research Article

PredDBP-Stack: Prediction of DNA-Binding Proteins from HMM Profiles using a Stacked Ensemble Method

Jun Wang,[1] **Huiwen Zheng**,[2] **Yang Yang**,[3] **Wanyue Xiao**,[4] **and Taigang Liu**[1]

[1]*College of Information, Shanghai Ocean University, Shanghai 201306, China*
[2]*School of Engineering, University of Melbourne, Victoria 3010, Australia*
[3]*School of Information Management, Nanjing University, Nanjing 210023, China*
[4]*School of Information, Syracuse University, Syracuse, NY 13244, USA*

Correspondence should be addressed to Taigang Liu; tgliu@shou.edu.cn

Received 2 March 2020; Accepted 1 April 2020; Published 13 April 2020

Guest Editor: Quan Zou

DNA-binding proteins (DBPs) play vital roles in all aspects of genetic activities. However, the identification of DBPs by using wet-lab experimental approaches is often time-consuming and laborious. In this study, we develop a novel computational method, called PredDBP-Stack, to predict DBPs solely based on protein sequences. First, amino acid composition (AAC) and transition probability composition (TPC) extracted from the hidden markov model (HMM) profile are adopted to represent a protein. Next, we establish a stacked ensemble model to identify DBPs, which involves two stages of learning. In the first stage, the four base classifiers are trained with the features of HMM-based compositions. In the second stage, the prediction probabilities of these base classifiers are used as inputs to the meta-classifier to perform the final prediction of DBPs. Based on the PDB1075 benchmark dataset, we conduct a jackknife cross validation with the proposed PredDBP-Stack predictor and obtain a balanced sensitivity and specificity of 92.47% and 92.36%, respectively. This outcome outperforms most of the existing classifiers. Furthermore, our method also achieves superior performance and model robustness on the PDB186 independent dataset. This demonstrates that the PredDBP-Stack is an effective classifier for accurately identifying DBPs based on protein sequence information alone.

1. Introduction

DNA-binding proteins (DBPs) are fundamental in the process of composing DNA and regulating genes. They execute intercellular and intracellular functions such as transcription, DNA replication, recombination, modification, and other biological activities associated with DNA [1]. As the significant role of DBPs undertaken, it has become one of the hot research topics to effectively identify DBPs in the field of protein science. The past decade has witnessed tremendous progress in the DBP recognition, including experimental methods, and computational methods [2]. In the early researches, DBPs were detected by laborious experimental techniques such as filter binding assays, genetic analysis, X-ray crystallography, chromatin immune precipitation on microarrays, and nuclear magnetic resonance [3]. With the rapid development of high-throughput sequencing technology and growing extension of protein sequence data, more efficient and accurate machine learning (ML) methods are implemented and applied for the classification of DBPs [4, 5].

Feature encoding schemes and classification algorithms have great impacts on the performance of ML-based methods. Feature representation numerically formulates diverse-length protein sequences as fixed-length feature vectors, which could be categorized into structure-based models and sequence-based models. Structure-based methods rely on the structure information of proteins such as the spatial distribution, net charge, electrostatic potential, the dipole moment, and quadrupole moment tensors [6, 7]. However, the great difficulty of acquiring the high-resolution crystal structure of proteins and the insufficient quantity of proteins with known structure information heavily limit the use of structure-based predictors [8].

In contrast, the sequence-based methods have become more popular since sequence features are usually easier to extract and more convenient to use. These sequence-based features of proteins are classified into three types: (1) composition-based features, such as amino acid composition (AAC) [9], dipeptide composition [10], and pseudo AAC [11–13]; (2) autocorrelation-based features, including autocross covariance [14, 15], normalized Moreau-Broto autocorrelation [8], and physicochemical distance transformation [16]; and (3) profile-based features, including position-specific score matrix (PSSM) [17–19] and hidden markov model (HMM) [20]. Generally, autocorrelation-based features perform better than composition-based features, and profile-based features outperform autocorrelation-based features [21].

Previous studies have demonstrated the importance of PSSM-based features for enhancing DBPs prediction. For example, Kumar et al. initially adopted evolutionary information embedded in the PSSM profile to identify DBPs and achieved a well-performed result [17]. Waris et al. produced an ingenious classifier by integrating the PSSM profile with dipeptide composition and split AAC [18]. Zou et al. proposed a fuzzy kernel ridge regression model to predict DBPs based on multiview sequence features [22]. Ali et al. introduced the DP-BINDER model for the discrimination of DBPs by fusing physicochemical information and PSSM-based features [23]. In the recent study, Zaman et al. built an HMMBinder predictor for the DBP recognition problem by extracting monogram and bigram features derived from the HMM profile [20]. They also experimentally proved that the HMM-based features are more effective for the prediction of DBPs than the PSSM-based features, especially on the jackknife test. Nevertheless, HMMBinder achieved relatively poor performance on the independent test. Accordingly, there is still more scope to improve the DBP prediction by exploring highly recognizable features from the HMM profile.

Prediction of DBPs is usually formulated as a supervised learning problem. In recent years, many classification algorithms have been adopted to solve this problem, including support vector machine (SVM) [24–26], random forest (RF) [27, 28], naive Bayes classifier [3], ensemble classifiers [29–31], and deep learning [32–34]. Among these models, stacked generalization (or stacking) is an ensemble learning technique that takes the outputs of base classifiers as input and attempts to find the optimal combination of the base learners to make a better prediction [35]. Xiong et al. constructed a stacked ensemble model to predict bacterial type IV secreted effectors from protein sequences by using the PSSM-composition features [36]. Recently, Mishra et al. developed a StackDPPred method for the effective prediction of DBPs, which utilized a stacking-based ML method and features extracted from the PSSM profiles [29].

Inspired by the work of Zaman and Mishra, respectively, we propose a stacked ensemble method, called PredDBP-Stack, to further improve the performance of DBP prediction by exploring valuable features from the HMM profiles. First, we convert the HMM profiles into 420-dimensional feature vectors by fusing AAC and transition probability composition (TPC) features. Next, six types of ML algorithms are adopted to implement base classifiers in the first stage. Then, the optimal combination of base learners is searched, and the prediction probabilities of these selected base learners are used as inputs to the meta-classifier to make the final prediction in the second stage. Compared with existing state-of-the-art predictors, our method performs better on the jackknife cross validation as well as on the independent test.

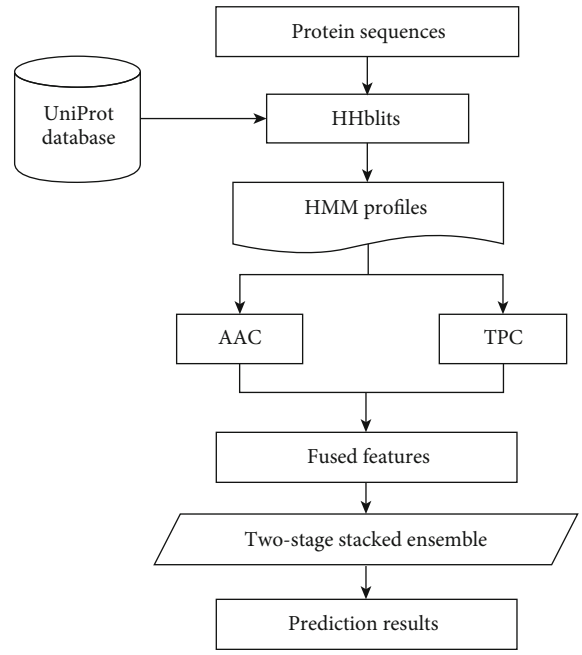

Figure 1: System diagram of PredDBP-Stack.

2. Materials and Methods

In this section, we describe all details about the proposed prediction model for identifying DBPs. The system diagram of the PredDBP-Stack methodology is illustrated in Figure 1. Several major intermediate steps in the development process of PredDBP-Stack are specified in the following subsections.

2.1. Datasets. The construction of a high-quality benchmark dataset is crucial for building a robust and reliable ML-based predictive model. In this study, two well-established datasets, i.e., PDB1075 [5] and PDB186 [3], are adopted to examine the performance of our predictor. The PDB1075 dataset consists of 1075 protein sequences with 525 DBPs and 550 non-DBPs, which are applied for model training and testing by using the jackknife cross validation. The PDB186 dataset is designed as an independent test dataset that contains 93 DBPs and 93 non-DBPs. All protein sequences in these two datasets were downloaded from the Protein Data Bank [37] and have been filtered rigorously by removing those with relatively high similarity (≥25%) or those with too small length (<50 amino acids) or involving unknown residues such as "X".

2.2. Feature Extraction

2.2.1. HMM Profiles. HMM profiles are supposed to contain rich evolution information of the query proteins and have

been widely used in bioinformatics, such as protein remote homology detection [38], DBP prediction [20], and protein fold recognition [39]. In this study, HMM profiles are generated from the multiple sequence alignments by running four iterations of the HHblits program [40] against the latest UniProt database [41] with default parameters. Similar to PSSM profile, we only use the first 20 columns of the HMM profile in the form of an $L \times 20$ matrix where L represents the length of the query protein sequence. Each element from the HMM profile is normalized by using the following function:

$$f(x) = \begin{cases} 0, & \text{if } x = *, \\ 2^{-x/1000}, & \text{else}, \end{cases} \quad (1)$$

where x is the original value of the HMM profile.

2.2.2. Feature Extraction from HMM Profiles. Feature extraction often plays an important role in most protein classification problems, which has a direct impact on the prediction accuracy of ML-based predictors. In this study, a simple and powerful feature encoding scheme by extracting AAC and TPC features is adopted to convert the HMM profiles into fixed-length feature vectors.

Since DNA-binding preference of a protein is closely related to its AAC [9], we first obtain AAC features from the HMM profile by using the following formula:

$$x_j = \frac{1}{L}\sum_{i=1}^{L} h_{i,j} \quad (j = 1, 2, \cdots, 20), \quad (2)$$

where $h_{i,j}$ is the value in the i^{th} row and j^{th} column of the HMM profile. x_j $(1 \le j \le 20)$ is the composition of amino acid type j in the HMM profile and represents the average score of the amino acid residues in the query protein being changed to amino acid type j during the evolution process. AAC based on the HMM profile is a simple and intuitive feature; however, it ignores the role of sequence-order information.

To partially reflect the local sequence-order effect, TPC features are computed from the HMM profile as follows:

$$y_{i,j} = \frac{\sum_{k=1}^{L-1} h_{k,i} \times h_{k+1,j}}{\sum_{j=1}^{20}\sum_{k=1}^{L-1} h_{k,i} \times h_{k+1,j}} \quad (1 \le i, j \le 20). \quad (3)$$

To include evolution information and sequence-order information, a 420-dimensional vector is finally employed to represent a protein by fusing AAC and TPC features. We call this feature encoding method AATP-HMM in this study.

2.3. Classification Algorithm. In this study, we apply one of the effective ensemble techniques called stacking [35] to achieve the performance improvement of the DBP predictor. Stacking makes up the limitation of the single classifier by integrating prediction results from multiple classification algorithms. There are two stages in our stacked ensemble scheme (Figure 2). For the first stage, various classification algorithms are employed individually as base classifiers to produce prediction class probabilities. For the second stage, these probabilities as inputs are taken into the meta-classifier in different combinations to generate desired prediction results.

To construct the well-behaved stacked model (SM) with the optimal combination of base classifiers, we explore six classification algorithms: (i) SVM with radial basis kernel function (RBF) [42], (ii) K Nearest Neighbor (KNN) [43], (iii) Logistic Regression (LR) [44], (iv) RF [45], (v) Decision Tree (DT) [46], and (vi) extreme Gradient Boosting (XGB) [47]. All of these algorithms are implemented by using scikit-learn library [48] in Python with the ideal parameters tuned based on the grid search strategy.

Taking into account the underlying principle of each classification algorithm and their prediction performance, we select three top learners, i.e., SVM (RBF), RF, and XGB, to, respectively, combine with other base classifiers. Also, we build the SM with these three best-performed classifiers and the one with all classification models. The following SMs are five combinations of base classifiers in this study:

(i) SM1: KNN, LR, DT, SVM (RBF)
(ii) SM2: KNN, LR, DT, XGB
(iii) SM3: KNN, LR, DT, RF
(iv) SM4: SVM (RBF), XGB, RF, and
(v) SM5: KNN, LR, DT, SVM (RBF), RF, XGB

In our stacked ensemble scheme, we adopt Gradient Boosting Decision Tree (GBDT) [49] as the meta-classifier to perform the final prediction of DBPs. Gradient boosting is a powerful ML technique, which produces a prediction model in the form of an ensemble of weak learners, typically DT [50]. Due to the arbitrary of choosing the loss function, GBDT could be customized to any particular ML task.

2.4. Performance Evaluation. To evaluate the performance of PredDBP-Stack, we first implement the jackknife cross-validation test on the PDB1075 dataset. In the jackknife test, every protein is tested one by one by the predictor trained with the remaining proteins in the benchmark dataset. Next, the independent test on the PDB186 dataset is also performed to examine the generalization ability of the proposed model. In this study, four widely used performance metrics are employed to compare PredDBP-Stack with several state-of-the-art models for identifying DBPs, including Overall Accuracy (OA), Sensitivity (SN), Specificity (SP), and Matthew's correlation coefficient (MCC) [51–54]. These metrics are formulated as follows:

$$\text{OA} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{FP} + \text{TN} + \text{FN}}, \quad (4)$$

$$\text{SN} = \frac{\text{TP}}{\text{TP} + \text{FN}}, \quad (5)$$

$$\text{SP} = \frac{\text{TN}}{\text{TN} + \text{FP}}, \quad (6)$$

$$\text{MCC} = \frac{\text{TP} \times \text{TN} - \text{FP} \times \text{FN}}{\sqrt{(\text{TP} + \text{FP}) \times (\text{TP} + \text{FN}) \times (\text{TN} + \text{FP}) \times (\text{TN} + \text{FN})}}, \quad (7)$$

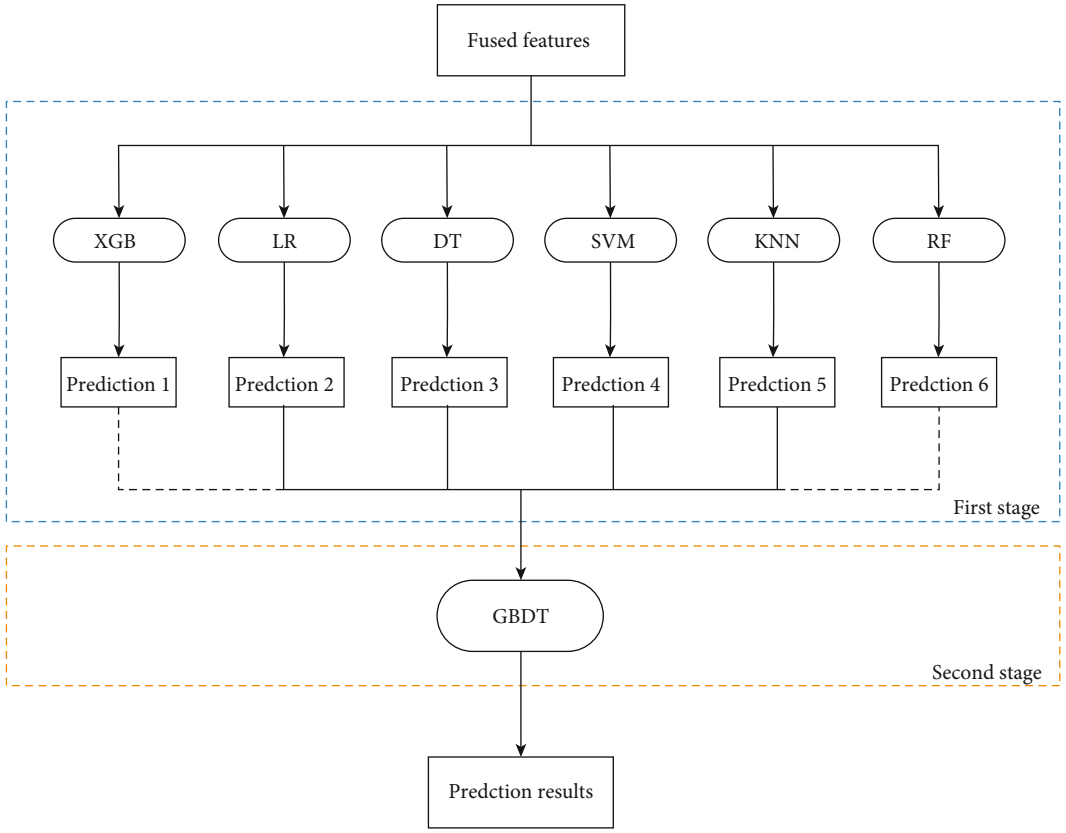

Figure 2: The framework of a two-stage stacked ensemble scheme.

where TN, FN, TP, and FP indicate the number of true negative, false negative, true positive, and false positive samples, respectively. Additionally, the area under the Receiver Operating Characteristic (ROC) Curve (AUC) is also computed as it is a powerful metric for evaluating the performance of a binary predictor. The larger the AUC value, the better the performance of the model.

3. Results and Discussion

3.1. Performance of Base Classifiers. Based on the AATP-HMM feature representation, we first analyze the predictive power of six classifiers, i.e., DT, KNN, LR, XGB, RF, and SVM employed in the base level of stacking. The models are tested on the PDB1075 dataset by using the jackknife cross validation and experimental results are shown in Table 1.

Table 1 indicates that the optimized SVM with RBF-kernel provides the highest performance in terms of OA, MCC, and AUC compared to the other methods for the prediction of DBPs. Moreover, the RF method obtains the best SN value of 83.4%, and the XGB method gives an outstanding SP value of 80.69%. It is also evident that the DT model performs worst in this task. In addition, the algorithms of KNN and LR show the acceptable performance with the AUC value larger than 0.8. To assure the distinct and high quality of the target figure, only three ROC curves corresponding with LR, DT, and SVM models are shown in Figure 3, which illustrates the consistent findings with Table 1.

Table 1: Performance comparison of six base classifiers on the PDB1075 dataset using the jackknife test.

Method	OA (%)	SN (%)	SP (%)	MCC	AUC
DT	74.53	74.71	74.36	0.4906	0.7838
KNN	76.22	75.68	76.73	0.5240	0.8364
LR	78.18	78.19	78.18	0.5635	0.8508
XGB	78.74	75.64	80.69	0.5634	0.8624
RF	78.28	83.4	73.45	0.5702	0.8648
SVM	80.34	81.66	79.27	0.6091	0.8774

3.2. Performance of Meta-Classifiers. To find out the optimal combination of base learners, we construct five SMs with different classifiers as follows. As SVM, XGB, and RF are the top three competitive classifiers in the above tests; each one of them is combined with the remaining classifiers to formulate an SM, namely SM1, SM2, and SM3, respectively. The combination of the three outstanding classifiers and all classifiers are formulated as SM4 and SM5. For all the SMs, the meta-classifier in the second stage is GBDT. The performance of five SMs on the PDB1075 dataset using the jackknife test is shown in Table 2.

From Table 2, we observe that SM1, SM2, SM3, and SM5 provide similar performance with the OA larger than 90%. However, SM4 produces less competitive scores on the five evaluation measures. It may imply that the combination of the top three competitive classifiers does not mean an

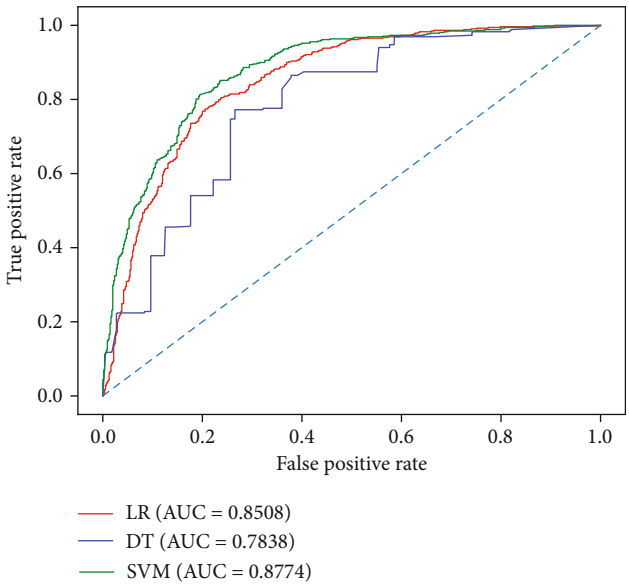

Figure 3: ROC curves of LR, DT, and SVM classifiers on the PDB1075 dataset using the jackknife test.

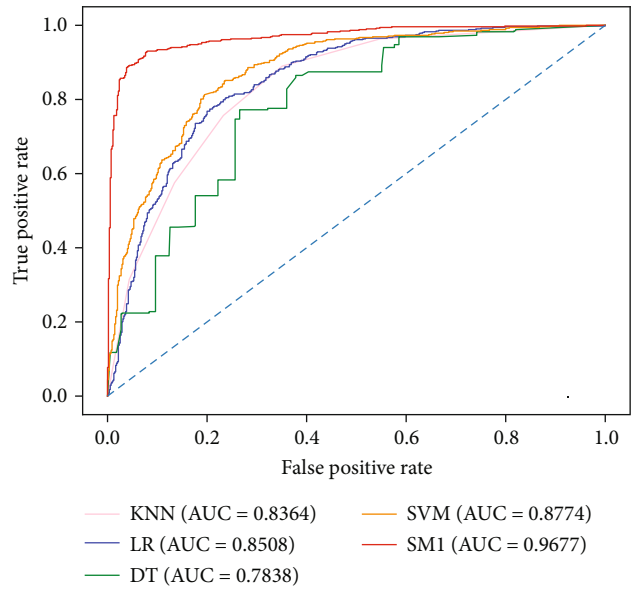

Figure 4: ROC curves of SM1 and its base classifiers on the PDB1075 dataset.

Table 2: Performance comparison of five SMs on the PDB1075 dataset using the jackknife test.

Method	OA (%)	SN (%)	SP (%)	MCC	AUC
SM1	92.42	92.47	92.36	0.8482	0.9677
SM2	92.23	91.89	92.55	0.8444	0.9664
SM3	91.76	91.31	92.18	0.8350	0.9635
SM4	79.87	82.82	77.09	0.5993	0.8745
SM5	90.54	90.93	90.18	0.8108	0.9560

advantageous result. Additionally, SM1, which employs KNN, LR, DT, and SVM (RBF) as base learners and GBDT as a meta-classifier, achieves the highest scores on the OA, SN, MCC, and AUC, respectively. SM2 gives the best SP of 92.55%. We also plot the ROC curves for SM1 and its four base classifiers in Figure 4, which demonstrates that stacked generalization can indeed improve the performance of base-level learners. Thus, SM1 is adopted as the final predictor for the identification of DBPs in the subsequent analysis.

3.3. Comparison with Existing Methods. In this section, we evaluate the performance of PredDBP-Stack by performing the following two testing protocols for a fair comparison with the existing methods, including DNABinder [17], DNA-Prot [4], iDNA-Prot [28], iDNA-Prot|dis [5], Kmer1+ACC [14], iDNAPro-PseAAC [19], Local-DPP [27], HMMBinder [20], and StackDPPred [29].

The jackknife test is first implemented on the benchmark dataset PDB1075, and the detailed results are reported in Table 3. As shown in Table 3, HMMBinder, StackDPPred, and the proposed PredDBP-Stack provide outstanding performance with the OA higher than 85% and the AUC value more than 0.9. However, our method shows the best predictive power on the five metrics: OA (92.42%), SN (92.47%), SP (92.36%), MCC (0.85), and AUC (0.9677). This is likely attributable to the effective feature extraction technique from the HMM profile and the powerful stacked ensemble classifier adopted in the PredDBP-Stack model.

Table 3: Performance comparison on the benchmark dataset PDB1075.

Method	OA (%)	SN (%)	SP (%)	MCC	AUC
DNA-Prot	72.55	82.67	59.76	0.44	0.7890
iDNA-Prot	75.40	83.81	64.73	0.50	0.7610
iDNA-Prot|dis	77.30	79.40	75.27	0.54	0.8260
DNABinder	73.95	68.57	79.09	0.48	0.8140
Kmerl+ACC	75.23	76.76	73.76	0.50	0.8280
iDNAPro-PseAAC	76.76	75.62	77.45	0.53	0.8392
Local-DPP	79.20	84.00	74.50	0.59	—
HMMBinder	86.33	87.07	85.55	0.72	0.9026
StackDPPred	89.96	91.12	88.80	0.80	0.9449
Our method	92.42	92.47	92.36	0.85	0.9677

To further assess the robustness of the proposed method, we perform an independent test on the PDB186 dataset, where PredDBP-Stack is beforehand trained on the benchmark dataset. Table 4 lists the predictive results of our method and nine existing state-of-the-art predictors mentioned above. From Table 4, we observe that our method, together with StackDPPred, performs better than the other methods on the PDB186 dataset, with the OA of 86.56%. Specifically, our method achieves the highest SP (86.02%) and AUC (0.8932) among the evaluated methods. In addition, the proposed PredDBP-Stack attains the second-best SN (87.10%) and MCC (0.731), which are

Table 4: Performance comparison on the independent dataset PDB186.

Method	OA (%)	SN (%)	SP (%)	MCC	AUC
DNA-Prot	61.80	69.90	53.80	0.240	0.7960
iDNA-Prot	67.20	67.70	66.70	0.334	0.8330
iDNA-Prot|dis	72.00	79.50	64.50	0.445	0.7860
DNABinder	60.80	57.00	64.50	0.216	0.6070
Kmerl+ACC	70.96	82.79	59.13	0.431	0.7520
iDNAPro-PseAAC	69.89	77.41	62.37	0.402	0.7754
Local-DPP	79.00	92.50	65.60	0.625	—
HMMBinder	69.02	61.53	76.34	0.394	0.6324
StackDPPred	86.55	92.47	80.64	0.736	0.8878
Our method	86.56	87.10	86.02	0.731	0.8932

slightly lower than those of StackDPPred. It should be pointed that the StackDPPred also applies a stacking technique to establish a powerful predictor for the identification of DBPs, which utilizes two different types of features, i.e., PSSM profile and residue wise contact energy profile [29]. However, our method also obtains favorable prediction accuracy when only the HMM profile is used. The successful applications of StackDPPred and PredDBP-Stack show that the stacking-based ML technique might yield a competitive tool for the prediction of DBPs and other protein classification tasks.

From the above comparisons, our method outperforms the existing models based on both the jackknife test and the independent test. This indicates that our method is a very promising tool for identifying DBPs and may at least play an important complementary role to existing methods.

4. Conclusions

Even though considerable efforts have been devoted so far, prediction of DBPs solely from sequence information still remains a challenging problem in bioinformatics. In this study, we develop a stacking-based ML model PredDBP-Stack to further improve prediction accuracy of DBPs, which employs an ensemble of base learners, such as KNN, LR, DT, and SVM, to generate outputs for the meta-classifier. Firstly, a hybrid feature encoding model, called AATP-HMM, is proposed to transform the HMM profiles to fixed-length numeric vectors, which incorporate evolution information and sequence-order effects. Next, these feature vectors are used to train the base-level predictors in the first stage. Then, GBDT is adopted as the meta-classifier in the second stage to implement the final prediction of DBPs. Finally, the jackknife cross validation and the independent test are performed on the two benchmark datasets to evaluate the predictive power of the proposed method. Comparison with the other existing predictors indicates that our method provides the outstanding improvement and could serve as a useful tool for predicting DBPs, given the sequence information alone.

Data Availability

The datasets and source codes for this study are freely available to the academic community at: https://github.com/taigangliu/PredDBP-Stack.

Conflicts of Interest

The authors declare that there is no conflict of interest regarding the publication of this paper.

Authors' Contributions

These authors contributed equally to this work as co-first authors.

Acknowledgments

The authors would like to thank Dr. Xiaoguang Bao for his pertinent suggestions. This work was funded by the National Natural Science Foundation of China (grant numbers 11601324, 11701363)

References

[1] R. E. Langlois and H. Lu, "Boosting the prediction and understanding of DNA-binding domains from sequence," *Nucleic Acids Research*, vol. 38, no. 10, pp. 3149–3158, 2010.

[2] K. Qu, L. Wei, and Q. Zou, "A review of DNA-binding proteins prediction methods," *Current Bioinformatics*, vol. 14, no. 3, pp. 246–254, 2019.

[3] W. Lou, X. Wang, F. Chen, Y. Chen, B. Jiang, and H. Zhang, "Sequence based prediction of DNA-binding proteins based on hybrid feature selection using random forest and Gaussian Naïve Bayes," *PLoS ONE*, vol. 9, no. 1, article e86703, 2014.

[4] K. K. Kumar, G. Pugalenthi, and P. N. Suganthan, "DNA-Prot: identification of DNA binding proteins from protein sequence information using random forest," *Journal of Biomolecular Structure & Dynamics*, vol. 26, no. 6, pp. 679–686, 2009.

[5] B. Liu, J. Xu, X. Lan et al., "iDNA-Prot|dis: identifying DNA-binding proteins by incorporating amino acid distance-pairs and reduced alphabet profile into the general pseudo amino acid composition," *PLoS ONE*, vol. 9, no. 9, article e106691, 2014.

[6] G. Nimrod, A. Szilágyi, C. Leslie, and N. Ben-Tal, "Identification of DNA-binding proteins using structural, electrostatic and evolutionary features," *Journal of Molecular Biology*, vol. 387, no. 4, pp. 1040–1053, 2009.

[7] G. Nimrod, M. Schushan, A. Szilágyi, C. Leslie, and N. Ben-Tal, "iDBPs: a web server for the identification of DNA binding proteins," *Bioinformatics*, vol. 26, no. 5, pp. 692-693, 2010.

[8] Y. Wang, Y. Ding, F. Guo, L. Wei, and J. Tang, "Improved detection of DNA-binding proteins via compression technology on PSSM information," *PLoS ONE*, vol. 12, no. 9, article e0185587, 2017.

[9] G. B. Motion, A. J. M. Howden, E. Huitema, and S. Jones, "DNA-binding protein prediction using plant specific support vector machines: validation and application of a new genome annotation tool," *Nucleic Acids Research*, vol. 43, no. 22, article e158, 2015.

[10] L. Nanni and A. Lumini, "Combing ontologies and dipeptide composition for predicting DNA-binding proteins," *Amino Acids*, vol. 34, no. 4, pp. 635–641, 2008.

[11] S. Adilina, D. M. Farid, and S. Shatabda, "Effective DNA binding protein prediction by using key features via Chou's general Pse AAC," *Journal of Theoretical Biology*, vol. 460, pp. 64–78, 2019.

[12] M. S. Rahman, S. Shatabda, S. Saha, M. Kaykobad, and M. S. Rahman, "DPP-Pse AAC: a DNA-binding protein prediction model using Chou's general Pse AAC," *Journal of Theoretical Biology*, vol. 452, pp. 22–34, 2018.

[13] X. Fu, W. Zhu, B. Liao, L. Cai, L. Peng, and J. Yang, "Improved DNA-binding protein identification by incorporating evolutionary information into the Chou's Pse AAC," *IEEE Access*, vol. 6, pp. 66545–66556, 2018.

[14] Q. Dong, S. Wang, K. Wang et al., "Identification of DNA-binding proteins by auto-cross covariance transformation," in *IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, pp. 470–475, Washington, DC, USA, 2015.

[15] B. Liu, S. Wang, Q. Dong, S. Li, and X. Liu, "Identification of DNA-binding proteins by combining auto-cross covariance transformation and ensemble learning," *IEEE Transactions on Nanobioscience*, vol. 15, no. 4, pp. 328–334, 2016.

[16] B. Liu, J. Xu, S. Fan, R. Xu, J. Zhou, and X. Wang, "Pse DNA-Pro: DNA-binding protein identification by combining Chou's Pse AAC and physicochemical distance transformation," *Molecular Informatics*, vol. 34, no. 1, pp. 8–17, 2015.

[17] M. Kumar, M. M. Gromiha, and G. P. Raghava, "Identification of DNA-binding proteins using support vector machines and evolutionary profiles," *BMC Bioinformatics*, vol. 8, no. 1, p. 463, 2007.

[18] M. Waris, K. Ahmad, M. Kabir, and M. Hayat, "Identification of DNA binding proteins using evolutionary profiles position specific scoring matrix," *Neurocomputing*, vol. 199, pp. 154–162, 2016.

[19] B. Liu, S. Wang, and X. Wang, "DNA binding protein identification by combining pseudo amino acid composition and profile-based protein representation," *Scientific Reports*, vol. 5, no. 1, article BFsrep15479, 2015.

[20] R. Zaman, S. Y. Chowdhury, M. A. Rashid et al., "HMMBinder: DNA-binding protein prediction using HMM profile based features," *Bio Med Research International*, vol. 4590609, p. 2017, 2017.

[21] J. Zhang and B. Liu, "A review on the recent developments of sequence-based protein feature extraction methods," *Current Bioinformatics*, vol. 14, no. 3, pp. 190–199, 2019.

[22] Y. Zou, Y. Ding, J. Tang, F. Guo, and L. Peng, "FKRR-MVSF: a Fuzzy Kernel Ridge Regression Model for identifying DNA-binding proteins by multi-view sequence features via Chou's five-step rule," *International Journal of Molecular Sciences*, vol. 20, no. 17, p. 4175, 2019.

[23] F. Ali, S. Ahmed, Z. N. K. Swati, and S. Akbar, "DP-BINDER: machine learning model for prediction of DNA-binding proteins by fusing evolutionary and physicochemical information," *Journal of Computer-Aided Molecular Design*, vol. 33, no. 7, pp. 645–658, 2019.

[24] L. Nanni and S. Brahnam, "Set of approaches based on 3D structure and position specific-scoring matrix for predicting DNA-binding proteins," *Bioinformatics*, vol. 35, no. 11, pp. 1844–1851, 2019.

[25] K. Qu, K. Han, S. Wu et al., "Identification of DNA-binding proteins using mixed feature representation methods," *Molecules*, vol. 22, no. 10, p. 1602, 2017.

[26] J. Zhang and B. Liu, "PSFM-DBT: identifying DNA-binding proteins by combing position specific frequency matrix and distance-bigram transformation," *International Journal of Molecular Sciences*, vol. 18, no. 9, p. 1856, 2017.

[27] L. Wei, J. Tang, and Q. Zou, "Local-DPP: an improved DNA-binding protein prediction method by exploring local evolutionary information," *Information Sciences*, vol. 384, pp. 135–144, 2017.

[28] W.-Z. Lin, J.-A. Fang, X. Xiao, and K. C. Chou, "iDNA-Prot: identification of DNA binding proteins using random forest with grey model," *PLoS ONE*, vol. 6, no. 9, p. e24756, 2011.

[29] A. Mishra, P. Pokhrel, and M. T. Hoque, "Stack DPPred: a stacking based prediction of DNA-binding protein from sequence," *Bioinformatics*, vol. 35, no. 3, pp. 433–441, 2019.

[30] X.-J. Liu, X.-J. Gong, H. Yu et al., "A model stacking framework for identifying DNA binding proteins by orchestrating multi-view features and classifiers," *Genes*, vol. 9, no. 8, p. 394, 2018.

[31] W. You, Z. Yang, G. Guo, X. F. Wan, and G. Ji, "Prediction of DNA-binding proteins by interaction fusion feature representation and selective ensemble," *Knowledge-Based Systems*, vol. 163, pp. 598–610, 2019.

[32] Y.-H. Qu, H. Yu, X.-J. Gong, J. H. Xu, and H. S. Lee, "On the prediction of DNA-binding proteins only from primary sequences: a deep learning approach," *PLoS ONE*, vol. 12, no. 12, article e0188129, 2017.

[33] S. Chauhan and S. Ahmad, "Enabling full-length evolutionary profiles based deep convolutional neural network for predicting DNA-binding proteins from sequence," *Proteins*, vol. 88, no. 1, pp. 15–30, 2020.

[34] S. Hu, R. Ma, and H. Wang, "An improved deep learning method for predicting DNA-binding proteins based on contextual features in amino acid sequences," *PLoS ONE*, vol. 14, no. 11, article e0225317, 2019.

[35] D. H. Wolpert, "Stacked generalization," *Neural Networks*, vol. 5, no. 2, pp. 241–259, 1992.

[36] Y. Xiong, Q. Wang, J. Yang, X. Zhu, and D. Q. Wei, "Pred T4SE-Stack: prediction of bacterial type IV secreted effectors from protein sequences using a stacked ensemble method," *Frontiers in Microbiology*, vol. 9, no. 2571, 2018.

[37] H. M. Berman, J. Westbrook, Z. Feng et al., "The protein data Bank," *Nucleic Acids Research*, vol. 28, no. 1, pp. 235–242, 2000.

[38] J. Chen, R. Long, X. L. Wang, B. Liu, and K. C. Chou, "dRHP-Pse RA: detecting remote homology proteins using profile-based pseudo protein sequence and rank aggregation," *Scientific Reports*, vol. 6, no. 1, article 32333, 2016.

[39] J. Lyons, K. K. Paliwal, A. Dehzangi, R. Heffernan, T. Tsunoda, and A. Sharma, "Protein fold recognition using HMM-HMM alignment and dynamic programming," *Journal of Theoretical Biology*, vol. 393, pp. 67–74, 2016.

[40] M. Remmert, A. Biegert, A. Hauser, and J. Söding, "HHblits: lightning-fast iterative protein sequence searching by HMM-HMM alignment," *Nature Methods*, vol. 9, no. 2, pp. 173–175, 2012.

[41] T. U. Consortium, "Uni Prot: the universal protein knowledgebase," *Nucleic Acids Research*, vol. 45, no. D1, pp. D158–D169, 2017.

[42] C. Cortes and V. Vapnik, "Support-vector networks," *Machine Learning*, vol. 20, no. 3, pp. 273–297, 1995.

[43] T. Cover and P. Hart, "Nearest neighbor pattern classification," *IEEE Transactions on Information Theory*, vol. 13, no. 1, pp. 21–27, 1967.

[44] D. W. Hosmer Jr., S. Lemeshow, and R. X. Sturdivant, *Applied logistic regression*, WILEY, 2013.

[45] H. Tin Kam, "Random decision forests," in *Proceedings of 3rd International Conference on Document Analysis and Recognition*, pp. 278–282, Montreal, Quebec, Canada, 1995.

[46] S. R. Safavian and D. Landgrebe, "A survey of decision tree classifier methodology," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 21, no. 3, pp. 660–674, 1991.

[47] T. Chen and C. Guestrin, "XGBoost: a scalable tree boosting system," in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 785–794, San Francisco, CA, USA, 2016.

[48] F. Pedregosa, G. Varoquaux, A. Gramfort et al., "Scikit-learn: machine learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[49] J. H. Friedman, "machine," *The Annals of Statistics*, vol. 29, no. 5, pp. 1189–1232, 2001.

[50] J. H. Friedman, "Stochastic gradient boosting," *Computational Statistics & Data Analysis*, vol. 38, no. 4, article S0167947301000652, pp. 367–378, 2002.

[51] Z. Lv, S. Jin, H. Ding, and Q. Zou, "A random forest sub-Golgi protein classifier optimized via dipeptide and amino acid composition features," *Frontiers in Bioengineering and Biotechnology*, vol. 7, p. 215, 2019.

[52] C. Meng, L. Wei, and Q. Zou, "Sec Pro MTB: support vector machine-based classifier for secretory proteins using imbalanced data sets applied to mycobacterium tuberculosis," *Proteomics*, vol. 19, no. 17, p. e1900007, 2019.

[53] C. Meng, S. Jin, L. Wang, F. Guo, and Q. Zou, "AOPs-SVM: a sequence-based classifier of antioxidant proteins using a support vector machine," *Frontiers in Bioengineering and Biotechnology*, vol. 7, p. 224, 2019.

[54] Z. Lv, J. Zhang, H. Ding, and Q. Zou, "RF-Pse U: a random forest predictor for RNA pseudouridine sites," *Frontiers in Bioengineering and Biotechnology*, vol. 8, p. 134, 2020.