Hindawi
BioMed Research International
Volume 2021, Article ID 5515342, 11 pages
https://doi.org/10.1155/2021/5515342

Research Article

i4mC-EL: Identifying DNA N4-Methylcytosine Sites in the Mouse Genome Using Ensemble Learning

Yanjuan Li, Zhengnan Zhao, and Zhixia Teng

College of Information and Computer Engineering, Northeast Forestry University, Harbin 150040, China

Correspondence should be addressed to Zhixia Teng; tengzhixia@nefu.edu.cn

Received 9 February 2021; Accepted 21 May 2021; Published 1 June 2021

Academic Editor: Andrea Scribante

As one of important epigenetic modifications, DNA N4-methylcytosine (4mC) plays a crucial role in controlling gene replication, expression, cell cycle, DNA replication, and differentiation. The accurate identification of 4mC sites is necessary to understand biological functions. In the paper, we use ensemble learning to develop a model named i4mC-EL to identify 4mC sites in the mouse genome. Firstly, a multifeature encoding scheme consisting of Kmer and EIIP was adopted to describe the DNA sequences. Secondly, on the basis of the multifeature encoding scheme, we developed a stacked ensemble model, in which four machine learning algorithms, namely, BayesNet, NaiveBayes, LibSVM, and Voted Perceptron, were utilized to implement an ensemble of base classifiers that produce intermediate results as input of the metaclassifier, Logistic. The experimental results on the independent test dataset demonstrate that the overall rate of predictive accurate of i4mC-EL is 82.19%, which is better than the existing methods. The user-friendly website implementing i4mC-EL can be accessed freely at the following.

1. Introduction

As a chemical modification occurring on DNA sequences, DNA methylation can change genetic properties under the condition that the order of DNA sequences remains unchanged. DNA methylation has many manifestations, such as 5-methylcytosine (5mC for short), N6-methyladenine (6 mA for short), and N4-methylcytosine (4mC for short) [1]. Among them, the 5mCs are widely present in prokaryotes and eukaryotes and are of great significance for controlling gene differentiation and gene expression, maintaining chromosome stability and cell structure [2, 3]. They also can cause some diseases such as cancer [4–6]. The 6mAs are also widely distributed in prokaryotes and eukaryotes, which play a crucial role in replication, expression, and transcription of gene [7]. The 4mCs which were found in 1983 mainly exist in prokaryotes, and they can control DNA replication, gene expression, and cell cycle [8]. However, compared with 5mCs and 6mAs, the current research on 4mCs is still insufficient. To make up for this defect and further understand 4mCs' biological properties and functions, the first thing we need to do is to identify 4mCs from various DNA sequences, which is still a hot research topic so far.

In order to identify 4mCs, many biology-based approaches have been explored. Single molecule real-time sequencing technology (SMRT for short) [9, 10] detects optical signals of bases matching the template at the single-molecule level to identify 4mCs. 4mC-Tet-assisted-bisulfite-sequencing technology (4mC-Tet for short) [11] identifies 4mCs by using bisulfite to convert unmethylated cytosine in the DNA sequences into uracil while to keep methylated cytosine unchanged. However, this kind of technologies is time-consuming and resource-intensive. Moreover, the explosive growth of DNA sequences also makes it more difficult to achieve whole-genome sequencing through these technologies. Therefore, using machine learning (ML for short) to identify 4mCs shows more advantages. Up to now, there are many models using machine learning to identify 4mCs. iDNA4mC [12], the earliest model for 4mC identification, is primarily used to identify 4mCs from the genomes of six species, A.thaliana, C.elegans, D.melanogaster, E.coli, G.pickeringii, and G.subterraneus, and its positive data containing 4mCs were obtained from a reliable

database called MethSMRT [13]. Soon afterwards, several other models, 4mcPred [14], 4mcPred-SVM [15], 4mcPred-IFL [16], and Meta-4mcPred [17], were proposed successively, which used the same dataset as iDNA4mC [12] for 4mCs identification of the genomes of these six species. i4mC-Rose [18] is the first and the only model for 4mCs identification in the genome of Rosaceae, and it derived positive dataset from the MDR [19] and the other reliable database for storing 4mC data. For the mouse genome we wanted to study, there have been currently two models, 4mCpred-EL [20] and i4mC-Mouse [21]. Among them, their samples containing 4mCs were also obtained from the MethSMRT database. In addition, 4mCpred-EL selected 4 ML algorithms and 7 feature encoding schemes to generate 28 sets of results as the final coding. Subsequently, 4mCpred-EL trained 4 submodels through the final coding and these 4 ML algorithms and then combined the 4 submodels into the final model by majority voting. i4mC-Mouse trained 6 submodels using 6 feature encoding schemes and random forest (RF for short) algorithm, and then the 6 submodels were combined into the final model by weighted voting. Compared with 4mCpred-EL, i4mC-mouse has better performance according to the indicators, ACC and MCC. Although exciting results have been achieved in 4mCpred-EL and i4mC-Mouse, the performance is able to be further increased. In this paper, to further improve the prediction capability, we propose a new mouse's 4mCs predictor, i4mC-EL.

2. Materials and Methods

2.1. Framework of i4mC-EL. In the present study, a novel model named i4mC-EL is proposed to indentify mouse's 4mCs, and we can see the framework of it in Figure 1. First, using two different feature encoding schemes, Kmer and EIIP, each DNA sequence was encoded into a 1364-dimensional vector and a 41-dimensional vector, respectively. Next, the 1364-dimensional vector and the 41-dimensional vector of each DNA sequence were combined to form a 1405-dimensional multifeature vector. Finally, a two-stage stacked ensemble learning classifier with these multifeature vectors as input was constructed. The ensemble classifier used BayesNet, NavieBayes Multinomial, LibSVM, and Voted Perceptron as base classifiers and used Logistic as metaclassifier. i4mC-EL's datasets and feature encoding schemes and classifiers will be described detailly below.

2.2. Dataset. This paper adopted the benchmark dataset constructed by Hasan'steam [21]. In this dataset, the positive samples containing mouse's 4mCs were obtained from the MethSMRT [17] database, and the negative samples were taken from chromosome DNA sequences. They were all fragments of DNA sequences consisting of 41 nucleotides with a "C" in the middle. Only the sequences whose modQV value greater than or equal to 20 were considered to obtain the high-quality dataset. To prevent the predictor from overfitting, the threshold of CD-HIT [22] was set to 70% to remove redundant sequences [23]. The dataset contained 1,812 DNA sequences, 906 of which were 4mCs and 906 were non-4mCs. About 80% of the dataset was randomly selected as the training dataset, and the remaining about 20% was used as the independent test dataset. The training dataset (train-1492) consisted of 746 4mCs and 746 non-4mCs. And the independent test dataset (test-320) included 160 4mCs and 160 non-4mCs.

2.3. Feature Encoding. Transforming DNA sequences into vectors that can make a distinction between 4mCs and non-4mCs availably is the first step to build an ensemble learning-based predictor to identify 4mCs [24–29]. Here, a multifeature encoding scheme composed of Kmer [30–33] and EIIP [34] was used to encode DNA sequences. Kmer represented the DNA sequences as the occurrence frequencies of k adjacent nucleotides. EIIP encoded each nucleotide in DNA sequences with its corresponding electron-ion energy. In the experiment of Section 3, we will find that this multifeature is able to encode DNA sequences availably. The following parts are detailed descriptions of Kmer and EIIP.

2.3.1. Kmer. This encoding scheme refers to the frequency of k-nucleotides composed of k continuous nucleotides in each sequence. For sequence $D = d_1 d_2 d_3 \cdots d_{L-2} d_{L-1} d_L$, each element of each feature vector is calculated by Equation (1):

$$f(X) = \frac{F(X)}{L - k + 1}, \tag{1}$$

where X is one of the k-nucleotide, $F(X)$and $f(X)$ are the count and frequency of X in D, respectively, and L is D's length. After Kmer, sequences are transformed into 4^k -dimensional vectors. For example, when the k-mer parameter $k = 2$, the value of AA in the 16-dimensional (4^2) feature vector of sequence D_1 = AAACTAGTC is 0.25.

In the present study, we choose the values of the parameter k to be 1, 2, 3, 4, and 5, generating 1364-dimensional ($4^1 + 4^2 + 4^3 + 4^4 + 4^5$) feature vectors.

2.3.2. EIIP. EIIP is the short name of electron-ion interaction pseudopotential. The encoding scheme based on EIIP was proposed by Nair and Sreenadhan in 2006. Through it, each nucleotide in each sequence is replaced by its corresponding electron-ion interaction pseud potential value (Table 1). For example, the result of sequence D_2 = AACTG after EIIP encoding is (0.1260, 0.1260, 0.1340, 0.1335, 0.0806). In the present study, each sequence is transformed into a 41-dimensional feature vector.

2.4. Classifier. As an open data mining platform, Weka has assembled a large number of machine learning algorithms that can undertake data mining tasks. In the present paper, the classifiers we used were all implemented by Weka, such as BayesNet, NaiveBayes, SGD, SimpleLogistic, SMO, IBk, JRip, J48, and ensemble learning. Finally, we chose the ensemble learning, and the results of related experiment will be presented in section 3.

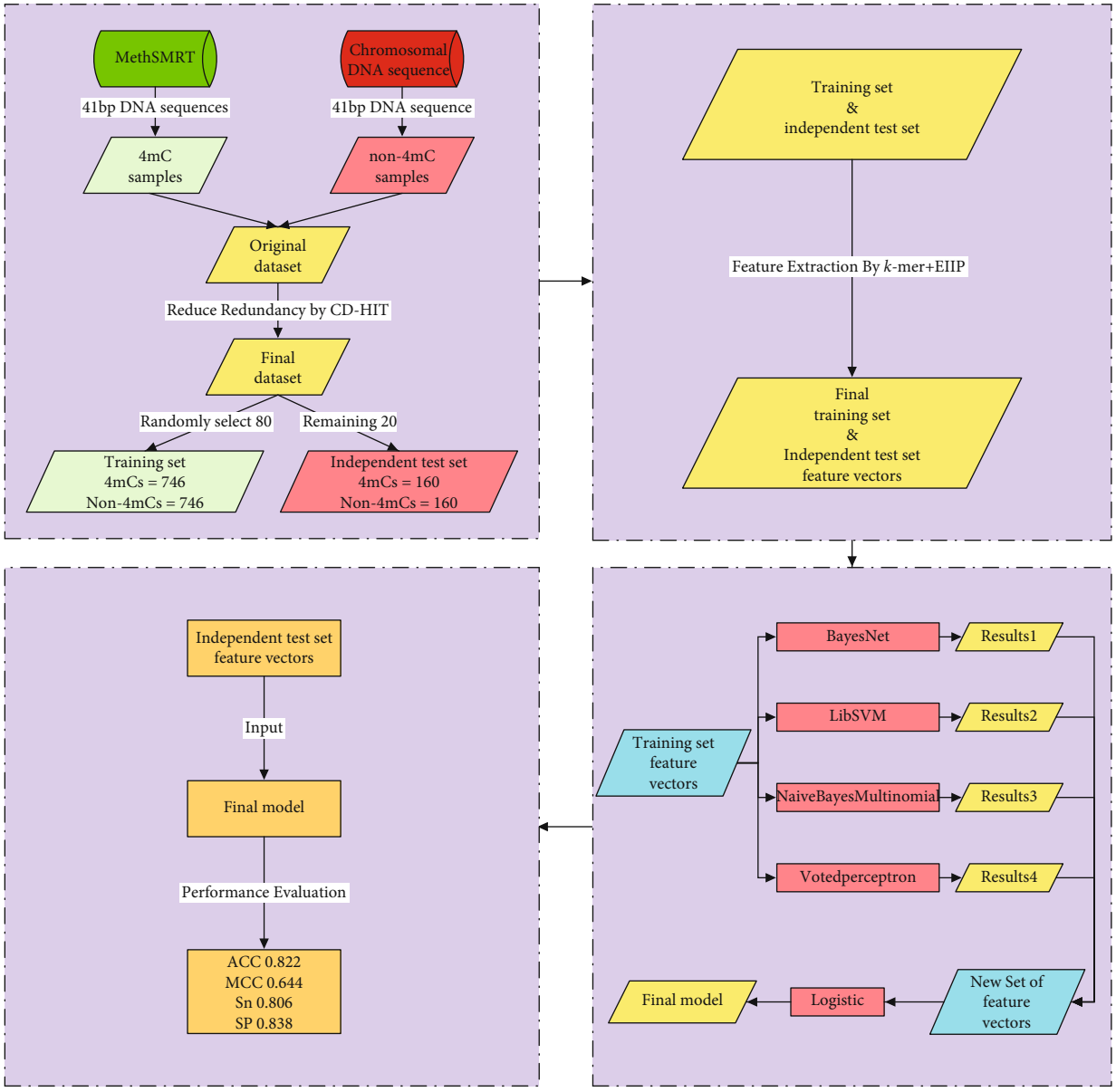

Figure 1: The framework of i4mC-EL.

Table 1: The electron-ion interaction pseudopotential values for DNA nucleotides.

NT	A	C	G	T
EIIP	0.1260	0.1340	0.0806	0.1335

According to different combination strategies, bagging, boosting, and stacking are the three main types of ensemble learning. Ensemble learning is widely used in bioinformatics because it can improve the prediction performance of classifiers, such as protein-protein interaction [35], disease prediction [36], type III secreted effectors prediction [37], and protein subcellular location prediction [38]. In detail, we used two-stage stacked ensemble learning.

In the two-stage stacked ensemble learning, the base classifiers used in this paper include BayesNet [39], Voted Perceptron [40], Naive Bayes Multinomial [41], and LibSVM [42], and the metaclassifier was Logistic. At the first stage of the ensemble learning classifier, based on the multifeature vectors proposed in this paper, four base classifiers are, respectively, trained to relabel the training dataset and the independent test dataset. At the second stage, the outputs of base classifiers are utilized as input for the metaclassifier.

Figure 2 gives the detailed process of model generation and result output, the steps are as follows.

Step 1. Partition dataset. Divide the training dataset into ten parts and mark them as train 1, train 2, …, train 10. The independent test dataset remains unchanged.

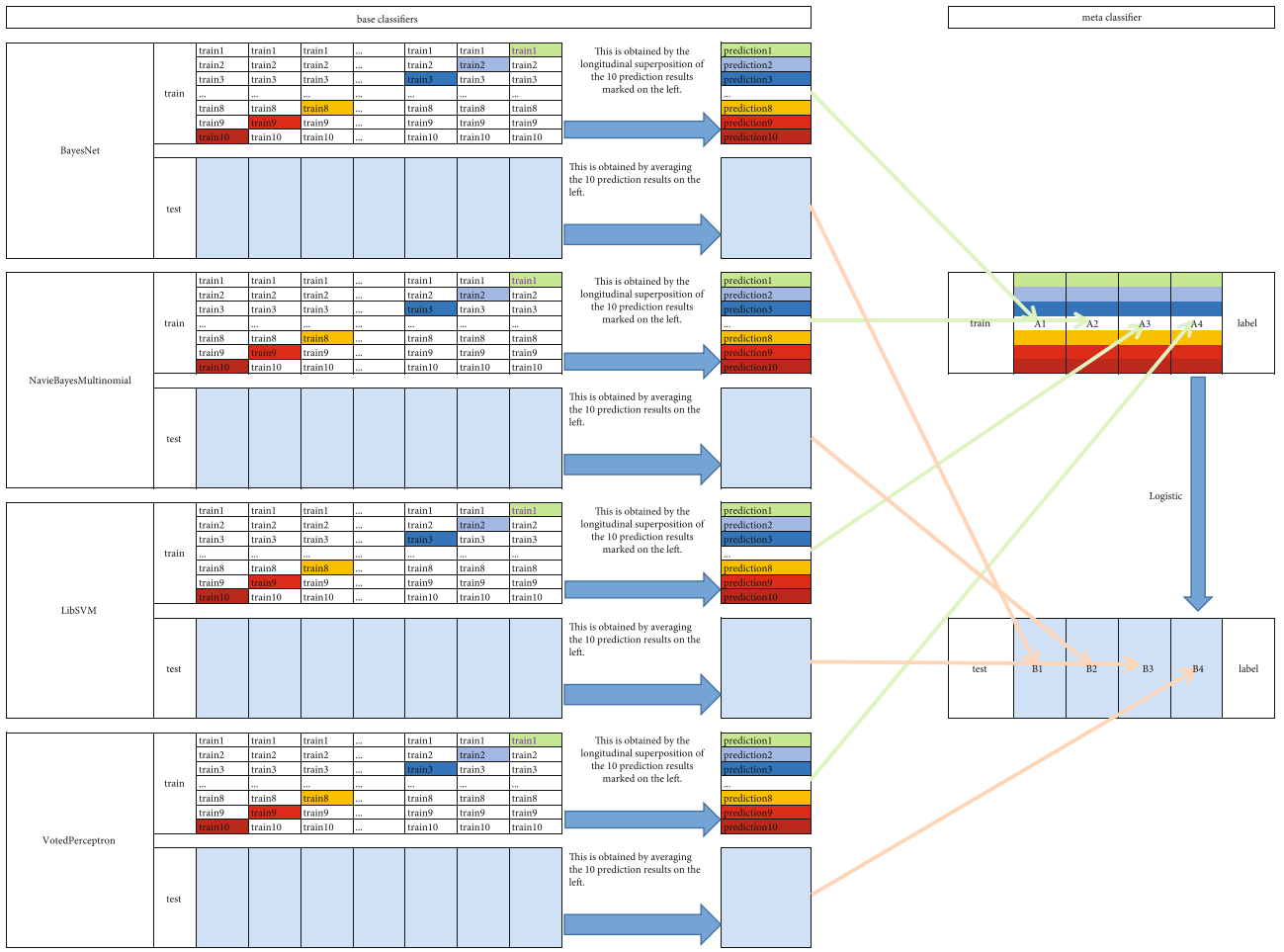

Figure 2: Working diagram of ensemble learning.

Step 2. Train base classifiers. In the present paper, we chose BayesNet, Voted Perceptron, Naive Bayes Multinomial, and LibSVM as base classifiers. For one base classifier such as BayesNet, 10-fold crossvalidation is performed. In detail, train 1, train 2, …, train 10 are used as validation dataset in turn, the other nine parts are used as the training dataset, and prediction is made on the independent test dataset. This would get 10 predictions from the training dataset together with another 10 predictions on the independent test dataset. Combine the 10 predictions on the training dataset vertically to get A1 and take the average of the 10 predictions on the independent test dataset to get B1. Similarly, we could get A2, B2 from NavieBayes Multinomial, A3, B3 from LibSVM, and A4, B4 from Voted Perceptron.

Step 3. Train metaclassifiers. Use the predictive values of the 4 base classifiers on the training dataset, A1, A2, A3, and A4, as 4 features to train the logistic classifier.

Step 4. Predict new data. Use the trained model to make predictions on the 4 features, B1, B2, B3, and B4, constructed from the predicted values of the independent test dataset of the 4 base classifiers, and then the final prediction results are obtained.

Table 2: The contrast of performance for dissimilar feature encoding schemes under 10-fold crossvalidation.

Schemes	ACC	MCC	Sn	Sp
BPF	0.668	0.335	0.665	0.670
DPE	0.614	0.228	0.619	0.609
RFHC	0.658	0.316	0.669	0.647
RevKmer	0.755	0.511	0.745	0.765
PseKNC	0.794	0.589	0.786	0.803
k-mer + BPF	0.724	0.448	0.729	0.718
k-mer + RFHC	0.747	0.493	0.744	0.749
RevKmer+DBE	0.738	0.476	0.723	0.753
RevKmer+EIIP	0.779	0.558	0.764	0.794
k-mer + BPF + DPE	0.732	0.464	0.741	0.723
Our method	0.803	0.606	0.784	0.822

2.5. Performance Evaluation. For the sake of validating the quality of our classification predictor, we used four indicators widely adopted in the field of bioinformatics for evaluation [43–53]. These indicators can be calculated using the

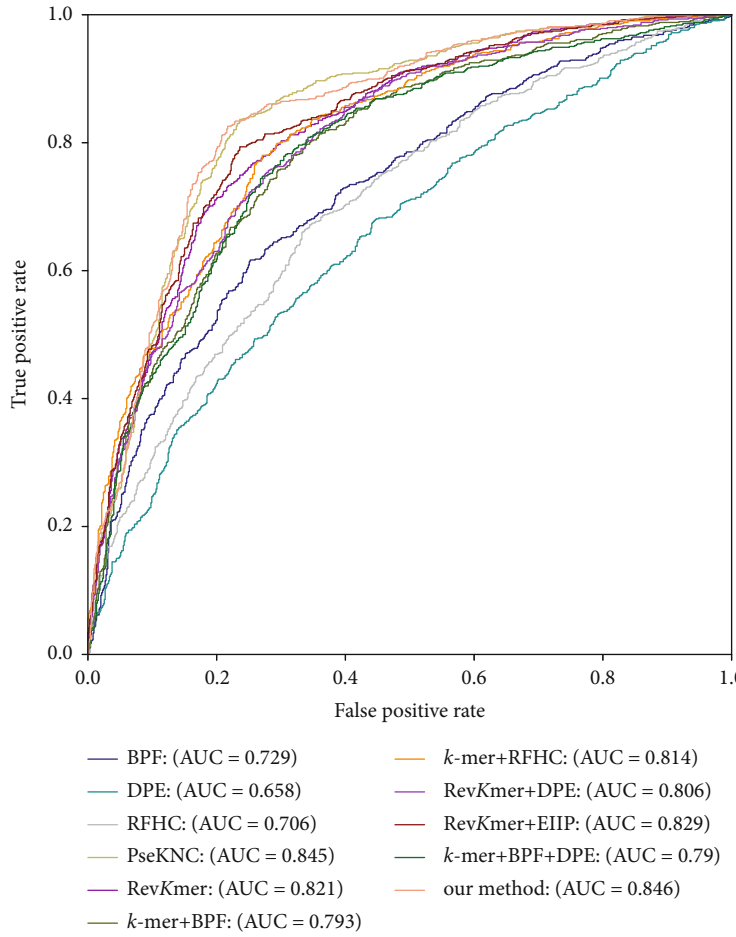

Figure 3: ROC curves for dissimilar feature encoding schemes under 10-fold crossvalidation.

formulas below:

$$\begin{aligned} \mathrm{ACC} &= \frac{\mathrm{TN}+\mathrm{TP}}{\mathrm{TN}+\mathrm{FN}+\mathrm{FP}+\mathrm{TP}}, \\ \mathrm{MCC} &= \frac{\mathrm{TN}\times\mathrm{TP}-\mathrm{FN}\times\mathrm{FP}}{\sqrt{(\mathrm{TN}+\mathrm{FN})\times(\mathrm{FN}+\mathrm{TP})\times(\mathrm{TP}+\mathrm{FP})\times(\mathrm{FP}+\mathrm{TN})}}, \\ \mathrm{Sn} &= \frac{\mathrm{TP}}{\mathrm{FN}+\mathrm{TP}}, \\ \mathrm{Sp} &= \frac{\mathrm{TN}}{\mathrm{FP}+\mathrm{TN}}, \end{aligned} \tag{2}$$

where TP indicates the number of the sequences that they are actually 4mCs, and that they are identified as 4mCs by the model, FP indicates the number of the sequences that they are actually non-4mCs but that they are identified as 4mCs by the model, TN indicates the number of the sequences that they are actually non-4mCs, and that they are identified as non-4mCs by the model, FN indicates the number of the sequences that they are actually 4mCs but that they are identified as non-4mCs by the model. The Sn refers to the prediction accuracy of 4mCs. The Sp refers to the prediction accuracy of non-4mCs. ACC refers to the prediction accuracy of both 4mCs and non-4mCs. MCC represents the reliability of the prediction results. The higher the values of the above four indicators have, the more superior the capability of the predictor would be.

Table 3: The contrast of performance for dissimilar classifiers under 10-fold crossvalidation.

Classifiers	ACC	MCC	Sn	Sp
BayesNet	0.727	0.453	0.739	0.714
NaiveBayes	0.752	0.504	0.751	0.753
SGD	0.712	0.424	0.710	0.713
SimpleLogistic	0.761	0.522	0.753	0.768
SMO	0.702	0.405	0.706	0.698
IBk	0.637	0.276	0.584	0.690
JRip	0.707	0.414	0.692	0.723
J48	0.665	0.330	0.674	0.655
RandomForest	0.770	0.541	0.753	0.787
AdaBoostM1	0.713	0.427	0.739	0.688
Bagging	0.729	0.459	0.744	0.714
Our method	0.803	0.606	0.784	0.822

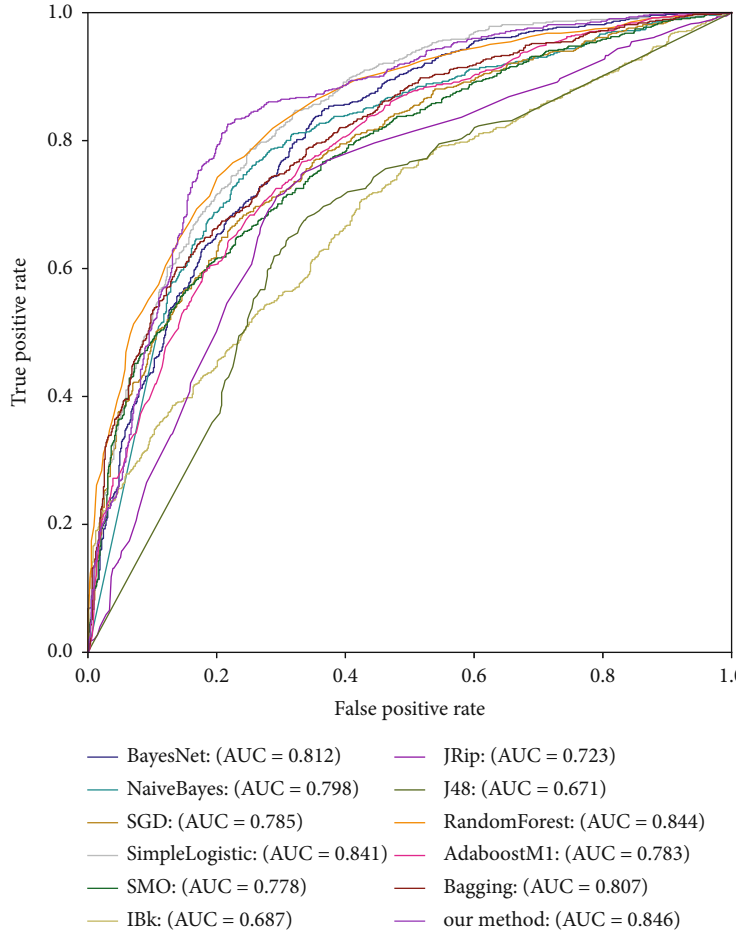

Figure 4: ROC curves for dissimilar classifiers under 10-fold crossvalidation.

3. Results and Discussion

3.1. Crossvalidation Results of TRAIN-1492. To find the features that can adequately represent the structure and function of the DNA sequences, we attempted to contrast numerous feature encoding schemes. And to achieve the optimal accuracy, we also tried to train the model using several different classification algorithms. The results of relevant comparative experiments are as below.

3.1.1. Feature Encoding Comparison on Crossvalidation. AS shown in section of "feature encoding," we encode the DNA sequences with a multifeature, which combines k-mer and EIIP feature encoding method. To verify the validity of the proposed multifeature, we compare the proposed multifeature with BPF, DPE, RFHC, RevKmer, and PseKNC feature encoding schemes and their combinations using ensemble learning classification. Among them, BPF and DPE are encoding schemes based on nucleotide positions, in which BPF takes mononucleotides as its encoding targets, while DPE takes dinucleotides as its encoding targets. RFHC is an encoding scheme based on the physicochemical properties of nucleotides. RevKmer is a variant of Kmer that considers not only the current k-nucleotides themselves, but also their reverse complementary nucleotides. PseKNC is a method to integrate continuous local and global k-tuple nucleotide information into the feature vectors of DNA sequences.

Table 4: The contrast of performance for dissimilar feature encoding schemes on TEST-320.

Schemes	ACC	MCC	Sn	Sp
BPF	0.753	0.530	0.606	0.900
DPE	0.697	0.401	0.600	0.794
RFHC	0.716	0.438	0.631	0.800
RevKmer	0.666	0.335	0.744	0.588
PseKNC	0.781	0.563	0.788	0.775
k-mer + BPF	0.772	0.553	0.681	0.863
k-mer + RFHC	0.800	0.614	0.694	0.906
RevKmer+DBE	0.756	0.516	0.700	0.813
RevKmer+EIIP	0.713	0.427	0.763	0.663
k-mer + BPF + DPE	0.772	0.553	0.681	0.863
Ourmethod	0.822	0.644	0.806	0.838

Table 2 displays experimental results, in which "our method" denotes the multifeature mentioned in the section "feature encoding." As shown in Table 2, from the perspective of ACC and MCC, the index values of our method are

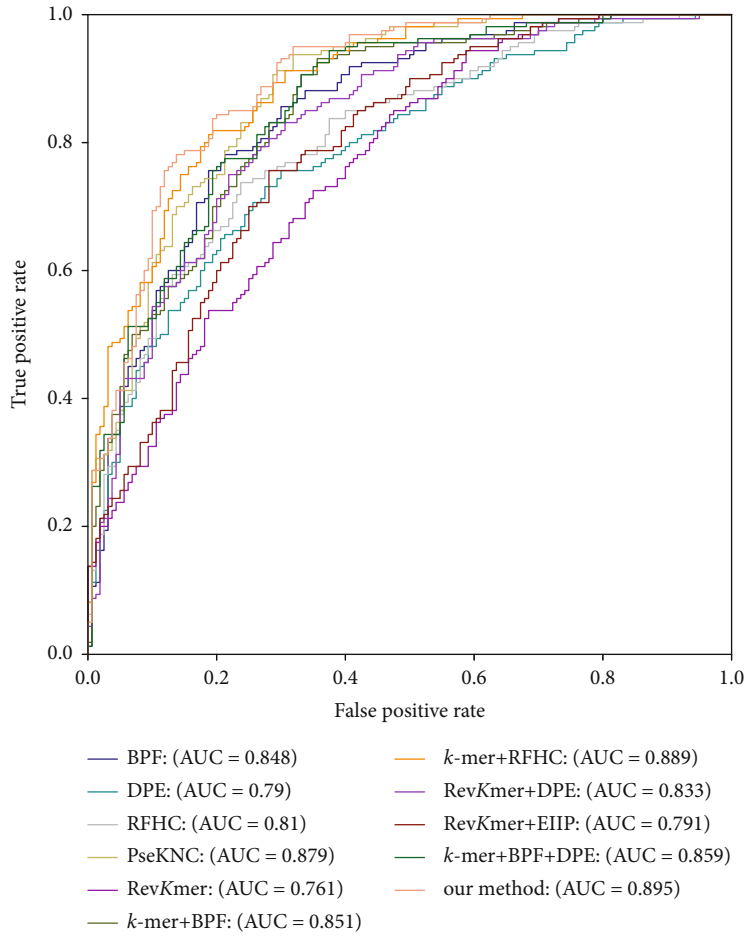

Figure 5: ROC curves for dissimilar feature encoding schemes on TEST-320.

higher than those of all other feature encoding schemes, which indicates that our method has a better overall performance. From the perspective of Sp, the index value of our method is still the highest, which indicates that it is more dominant to identify non-4mC from negative samples. These conclusions demonstrate that our method has good validity.

To further illustrate the prediction capability of our selected multifeature encoding scheme, the ROC curves for dissimilar feature encoding schemes under 10-fold crossvalidation are displayed in Figure 3. From Figure 3, we can see that our method has the largest area under ROC curve (AUC), which demonstrates that our method can represent mouse's DNA sequences better than others.

3.1.2. Classifier Comparison on Crossvalidation. As shown in the section "classifier," we inputted the multifeature composed of *k*-mer and EIIP into an ensemble learning classifier called stacking, then obtained a predictor which is used for identifying mouse's 4mCs. To verify the validity of stacking used in this paper, on the basis of the multifeature used in this paper, we compared stacking with eleven commonly used classifiers, BayesNet, Naive Bayes, SGD, Simple Logistic, SMO, IBK, JRip, J48, Random Forest, AdaBoostM1, and Bagging. Among them, BayesNet characterizes the dependencies among attributes with the aid of directed acyclic graphs and uses conditional probability tables to describe the joint probability distribution of attributes. NaiveBayes is a simple probabilistic classifier based on Bayes' theorem under the assumption that each attribute is independent of each other. SGD implements a regularized linear support vector machine classifier with stochastic gradient descent learning. Simple

Table 5: The contrast of performance for dissimilar classifiers on TEST-320.

Classifiers	ACC	MCC	Sn	Sp
BayesNet	0.769	0.547	0.675	0.863
NaiveBayes	0.788	0.577	0.744	0.831
SGD	0.688	0.379	0.756	0.619
Simple Logistic	0.728	0.456	0.738	0.719
SMO	0.675	0.353	0.744	0.606
IBk	0.600	0.201	0.563	0.638
JRip	0.769	0.541	0.713	0.825
J48	0.663	0.325	0.656	0.669
Random Forest	0.778	0.558	0.738	0.819
AdaBoostM1	0.791	0.581	0.794	0.788
Bagging	0.781	0.564	0.744	0.819
Our method	0.822	0.644	0.806	0.838

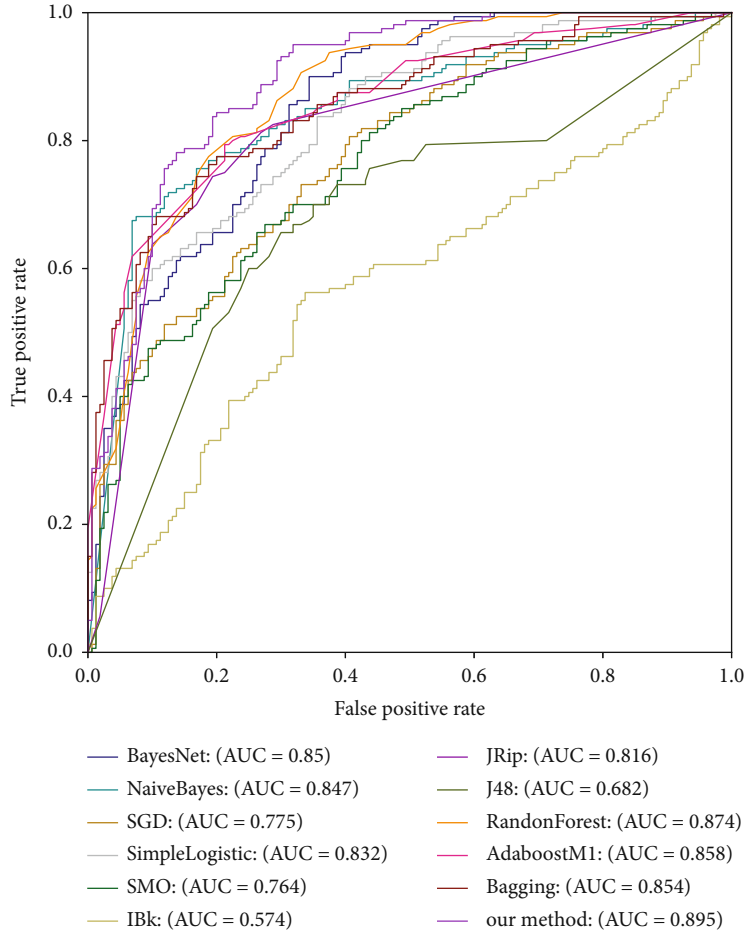

Figure 6: ROC curves for dissimilar classifiers on TEST-320.

Logistic is a linear logistic regression classifier with only one independent variable. SMO is a support vector machine classifier using a continuous minimum optimization algorithm. IBk classifies the data point by determining the category of k data points closest to it. JRip is a classifier based on rule induction. J48 is a decision tree classifier that uses information gain rate to select attributes for partitioning. Random Forest refers to a classifier that utilizes multiple trees to train and predict a sample. AdaBoostM1 is a classifier that enables the previously incorrectly predicted training samples to receive more attention at follow-up by adjusting their distribution. Bagging uses bootstrap sampling to obtain m (m is the predetermined number of base classifiers) sample datasets from the original dataset, which are used to train m base classifiers that are then integrated by voting.

The results of these comparative experiments are displayed in Table 3, where "our method" refers to the stacking classifier. From Table 3, we can see that our method outperforms the other classifiers in all indicators.

To further illustrate the classification capability of our selected stacking classifier, the ROC curves for dissimilar classifiers under 10-fold crossvalidation are displayed in Figure 4. From Figure 4, we can see that the area under ROC curve (AUC) of our method is the largest, which proves that our proposed method has better prediction performance for identifying 4mCs in the mouse genome than other methods.

3.2. Independent Validation Results of TEST-320. In this section, a comparative experiment on the independent test dataset (TEST-320) will be conducted to show the generalization capability of our selected multifeature and stacking classifier. The rationale for this is that this model is trained and tested on two different datasets, which is the equivalent of performing a real prediction task with the generated model.

3.2.1. Feature Encoding Comparison on Independent Validation. Using the stacking classifier, we, respectively, evaluate the generalization capability of various feature encoding schemes described in Section 3.1.1 on TEST-320. Table 4 displays these comparison experimental results. From Table 4, among the compared feature encoding schemes, our method performed best in ACC, Sn, and MCC, which were 82.19%, 0.806, and 0.644, respectively. Although the Sp of our method is lower than that of BPF, k-mer + BPF, k-mer + RFHC, k-mer + BPF + DPE, and PseKNC+EIIP+RFHC, the other three indicators of our method are higher than theirs.

For the sake of further describing the generalization capability of our selected multifeature encoding scheme, Figure 5

Table 6: The contrast of performance for dissimilar models on TEST-320.

Models	ACC	MCC	Sn	Sp
4mcPred-EL	0.791	0.584	0.757	0.825
i4mC-Mouse	0.816	0.633	0.807	0.825
i4mC-EL	0.822	0.644	0.806	0.838

displays the ROC curves for dissimilar feature encoding schemes on TEST-320. From Figure 5, we can see that the AUC of our method is the largest, and the ROC curve of our method is closer to the upper left, which demonstrates that our selected multifeature is more suitable than other schemes to encode the DNA sequences used to recognize mouse's 4mC.

3.2.2. Classifier Comparison on Independent Validation. We compared stacking classifier used in this paper with other eleven classifiers on TEST-320 under the condition of using the multifeature combing *k*-mer and EIIP as the input of the stacking. The results of these comparative experiments are displayed in Table 5, from which we can see that although the Sp of BayesNet is a little higher than that of our method, our method outperforms other classifiers in ACC, Sn, and MCC. Overall, our selected stacking classifier performs better than the others, indicating that it is effective for identifying mouse's 4mC.

For the sake of further describing the generalization capability of our selected stacking classifier, the ROC curves for dissimilar classifiers on TEST-320 are displayed in Figure 6, where we can get the conclusion that the AUC of our method is the largest too, which proves that our proposed stacking-based ensemble classifier method is more suitable for the identification of mouse's 4mCs than other classifiers.

3.3. Contrast with Extant Models on TEST-320. Here, we contrasted i4mC-EL with 4mCpred-EL and i4mC-Mouse on TEST-320 for the sake of further evaluating its performance. Table 6 displays these contrast experimental results, in which the data of 4mCpred-EL and i4mC-Mouse are from reference. From Table 6, we can see that i4mC-EL is superior to 4mcPred-EL and i4mC-Mouse in three indexes which are ACC, Sp, and MCC. Although the Sn of i4mC-Mouse is a little higher than that of our method, our method outperforms i4mC-Mouse in the other three indexes. All in all, i4mC-EL performs better than extant methods.

4. Conclusions

In the present paper, an ensemble learning model called i4mC-EL which was able to identify mouse's 4mC sites was designed. In the process of constructing i4mC-EL, to determine the optimal combination of feature encoding schemes and classifiers, we conducted abundant comparative experiments on dissimilar features and classifiers. Finally, we encoded DNA sequences with multifeatures combing *k*-mer and EIIP, then used two-stage stacked ensemble learning as classifier. We used BayesNet, NavieBayes Multinomial, LibSVM, and VotedPerceptron as base classifiers and Logistic as metaclassifier.

In addition, we contrasted i4mC-EL with existing models for the sake of proving its effectiveness. The results show that i4mC-EL is better than the existing models and has better generalization capability. In summary, i4mC-EL is effective in predicting the 4mC sites in the mouse genome, which helps us to understand the biochemical properties of 4mC.

We will use adaptive feature vectors to donate DNA sequences to optimize the feature encoding scheme [54, 55] in the future work. Furthermore, other improvements, encoding schemes, classifier algorithms, and intelligent computing models to identify 4mC sites will also be considered.

Data Availability

The datasets used during the present study are available from the corresponding author upon reasonable request, or can be downloaded from http://106.12.83.135:8080/i4mC-EL/.

Conflicts of Interest

The authors declare that they have no competing interests.

Acknowledgments

This work is supported by the Natural Science Foundation of Heilongjiang Province (LH2019F002), National Natural Science Foundation of China (61901103, 61671189), and the Fundamental Research Funds for the Central Universities (2572018BH05, 2572017CB33).

References

[1] H. Lv, F. Y. Dao, D. Zhang et al., "iDNA-MS: An integrated computational tool for detecting DNA modification sites in multiple genomes," *iScience*, vol. 23, no. 4, article 100991, 2020.

[2] S. M. Irene, S. Maria, M. M. Rosaria, and D. E. Maurizio, "DNA methylation 40 years later: its role in human health and disease," *Journal of Cellular Physiology*, vol. 204, no. 1, pp. 21–35, 2005.

[3] Y. Zuo, M. Song, H. Li et al., "Analysis of the epigenetic signature of cell reprogramming by computational DNA methylation profiles," *Current Bioinformatics*, vol. 15, no. 6, pp. 589–599, 2020.

[4] C. Ling and L. Groop, "Epigenetics: a molecular link between environmental factors and type 2 Diabetes," *Diabetes*, vol. 58, no. 12, pp. 2718–2725, 2009.

[5] Y. Zhang, C. Kou, S. Wang, and Y. Zhang, "Genome-wide differential-based analysis of the relationship between DNA methylation and gene expression in cancer," *Current Bioinformatics*, vol. 14, no. 8, pp. 783–792, 2019.

[6] W. Tang, S. Wan, Z. Yang, A. E. Teschendorff, and Q. Zou, "Tumor origin detection with tissue-specific miRNA and DNA methylation markers," *Bioinformatics*, vol. 34, no. 3, pp. 398–406, 2018.

[7] Y. Fu, G.-Z. Luo, K. Chen et al., "N^6-Methyldeoxyadenosine Marks Active Transcription Start Sites in *Chlamydomonas*," *Cell*, vol. 161, no. 4, pp. 879–892, 2015.

[8] K. Chen, B. S. Zhao, and C. He, "Nucleic acid modifications in regulation of gene expression," *Cell Chemical Biology*, vol. 23, no. 1, pp. 74–85, 2016.

[9] B. A. Flusberg, D. R. Webster, J. H. Lee et al., "Direct detection of DNA methylation during single-molecule, real-time sequencing," *Nature Methods*, vol. 7, no. 6, pp. 461–465, 2010.

[10] T. Zhu, J. Guan, H. Liu, and S. Zhou, "RMDB: an integrated database of single-cytosine-resolution DNA methylation in Oryza sativa," *Current Bioinformatics*, vol. 14, no. 6, pp. 524–531, 2019.

[11] Y. Huang, H.-T. Ren, Q. Zou, Y.-Q. Wang, J.-L. Zhang, and X.-L. Yu, "Computational identification and characterization of miRNAs and their target genes from five cyprinidae fishes," *Saudi Journal of Biological Sciences*, vol. 246, pp. 1126–1135, 2017.

[12] C. Wei, Y. Hui, F. Pengmian, D. Hui, and L. Hao, "iDNA4mC: identifying DNA N4-methylcytosine sites based on nucleotide chemical properties," *Bioinformatics*, vol. 33, no. 22, pp. 3518–3523, 2017.

[13] P. Ye, Y. Luan, K. Chen, Y. Liu, C. Xiao, and Z. Xie, "MethSMRT: an integrative database for DNA N6-methyladenine and N4-methylcytosine generated by single-molecular real-time sequencing," *Nucleic Acids Research*, vol. 45, no. D1, pp. D85–D89, 2017.

[14] H. Wenying, J. Cangzhi, and Z. Quan, "4mCPred: machine learning methods for DNA N4-methylcytosine sites prediction," *Bioinformatics*, vol. 35, no. 4, pp. 593–601, 2019.

[15] W. Leyi, L. Shasha, N. L. A. Eijy, S. Ran, and Z. Quan, "Exploring sequence-based features for the improved prediction of DNA N4-methylcytosine sites in multiple species," *Bioinformatics*, vol. 35, no. 8, pp. 1326–1333, 2019.

[16] W. Leyi, S. Ran, L. Shasha et al., "Iterative feature representations improve N4-methylcytosine site prediction," *Bioinformatics*, vol. 35, no. 23, pp. 4930–4937, 2019.

[17] B. Manavalan, S. Basith, T. H. Shin, L. Wei, and G. Lee, "Meta-4mCpred: a sequence-based meta-predictor for accurate DNA 4mC site prediction using effective feature representation," *Molecular Therapy - Nucleic Acids*, vol. 16, pp. 733–744, 2019.

[18] M. M. Hasan, B. Manavalan, M. S. Khatun, and H. Kurata, "i4mC-ROSE, a bioinformatics tool for the identification of DNA N4-methylcytosine sites in the Rosaceae genome," *International Journal of Biological Macromolecules*, vol. 157, pp. 752–758, 2020.

[19] Z.-Y. Liu, J.-F. Xing, W. Chen et al., "MDR: an integrative DNA N6-methyladenine and N4-methylcytosine modification database for Rosaceae," *Horticulture Research*, vol. 6, no. 1, 2019.

[20] B. Manavalan, S. Basith, T. H. Shin, D. Y. Lee, L. Wei, and G. Lee, "4mCpred-EL: An Ensemble Learning Framework for Identification of DNA N4-Methylcytosine Sites in the Mouse Genome," *Cell*, vol. 8, no. 11, p. 1332, 2019.

[21] M. M. Hasan, B. Manavalan, W. Shoombuatong, M. S. Khatun, and H. Kurata, "i4mC-mouse: improved identification of DNA N4-methylcytosine sites in the mouse genome using multiple encoding schemes," *Computational and Structural Biotechnology Journal*, vol. 18, pp. 906–912, 2020.

[22] L. Fu, B. Niu, Z. Zhu, S. Wu, and W. Li, "CD-HIT: accelerated for clustering the next-generation sequencing data," *Bioinformatics*, vol. 28, no. 23, pp. 3150–3152, 2012.

[23] Q. Zou, G. Lin, X. Jiang, X. Liu, and X. Zeng, "Sequence clustering in bioinformatics: an empirical study," *Briefings in Bioinformatics*, vol. 21, no. 1, pp. 1–10, 2020.

[24] Y. Zhang and Q. Zou, "PPTPP: a novel therapeutic peptide prediction method using physicochemical property encoding and adaptive feature representation learning," *Bioinformatics*, vol. 36, no. 13, pp. 3982–3987, 2020.

[25] 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.

[26] Q. Zou, P. Xing, L. Wei, and B. Liu, "Gene2vec: gene subsequence embedding for prediction of mammalianN6-methyladenosine sites from mRNA," *RNA*, vol. 25, no. 2, pp. 205–218, 2019.

[27] B. Liu, X. Gao, and H. Zhang, "BioSeq-Analysis2.0: an updated platform for analyzing DNA, RNA, and protein sequences at sequence level and residue level based on machine learning approaches," *Nucleic Acids Research*, vol. 47, no. 20, article e127, 2019.

[28] J. Shao, K. Yan, and B. Liu, "FoldRec-C2C: protein fold recognition by combining cluster-to-cluster model and protein similarity network," *Briefings in Bioinformatics*, vol. 22, no. 3, 2021.

[29] Y.-J. Tang, Y.-H. Pang, and B. Liu, "IDP-Seq2Seq: identification of intrinsically disordered regions based on sequence to sequence learning," *Bioinformaitcs*, vol. 36, no. 21, pp. 5177–5186, 2021.

[30] B. Liu, H. Wu, and K.-C. Chou, "Pse-in-One 2.0: an improved package of web servers for generating various modes of pseudo components of DNA, RNA, and protein sequences," *Natural Science*, vol. 9, no. 4, pp. 67–91, 2017.

[31] C.-C. Li and B. Liu, "MotifCNN-fold: protein fold recognition based on fold-specific features extracted by motif-based convolutional neural networks," *Briefings in Bioinformatics*, vol. 21, no. 6, pp. 2133–2141, 2020.

[32] Z. Y. Zhang, Y. H. Yang, H. Ding, D. Wang, W. Chen, and H. Lin, "Design powerful predictor for mRNA subcellular location prediction in Homo sapiens," *Briefings in Bioinformatics*, vol. 22, no. 1, pp. 526–535, 2021.

[33] H. Yang, W. Yang, F. Y. Dao et al., "A comparison and assessment of computational method for identifying recombination hotspots in Saccharomyces cerevisiae," *Briefings in Bioinformatics*, vol. 21, no. 5, pp. 1568–1580, 2020.

[34] A. S. Nair and S. S. Pillai, "A coding measure scheme employing electron-ion interaction pseudopotential (EIIP)," *Bioinformation*, vol. 1, no. 6, 2006.

[35] H. Zhu, X. Du, and Y. Yao, "ConvsPPIS: identifying protein-protein interaction sites by an ensemble convolutional neural network with feature graph," *Current Bioinformatics*, vol. 15, no. 4, pp. 368–378, 2020.

[36] N. Sultana, N. Sharma, K. P. Sharma, and S. Verma, "A sequential ensemble model for communicable disease forecasting," *Current Bioinformatics*, vol. 15, no. 4, pp. 309–317, 2020.

[37] J. Li, L. Wei, F. Guo, and Q. Zou, "EP3: An ensemble predictor that accurately identifies type III secreted effectors," *Briefings in Bioinformatics*, vol. 22, no. 2, pp. 1918–1928, 2020.

[38] S. Wan, Y. Duan, and Q. Zou, "HPSLPred: an ensemble multi-label classifier for human protein subcellular location prediction with imbalanced source," *Proteomics*, vol. 17, no. 17-18, p. 1700262, 2017.

[39] N. Friedman, D. Geiger, and M. Goldszmidt, "Bayesian Network Classifiers," *Machine Learning*, vol. 29, no. 2/3, pp. 131–163, 1997.

[40] Y. Freund and R. E. Schapire, "Large margin classification using the perceptron algorithm," *Machine Learning*, vol. 37, no. 3, 1999.

[41] M. C. Wang, "A prior-valued estimator applied to multinomial classification," *Communications in Statistics - Theory and Methods*, vol. 15, no. 2, pp. 405–427, 1986.

[42] C.-C. Chang and C.-J. Lin, "LIBSVM," *ACM Transactions on Intelligent Systems and Technology*, vol. 2, no. 3, pp. 1–27, 2011.

[43] Z. B. Lv, D. H. Wang, H. Ding, B. N. Zhong, and L. Xu, "Escherichia Coli DNA N-4-methycytosine site prediction accuracy improved by light gradient boosting machine feature selection technology," *IEEE Access*, vol. 8, pp. 14851–14859, 2020.

[44] J. Wang, Y. Shi, X. Wang, and H. Chang, "A drug target interaction prediction based on LINE-RF learning," *Current Bioinformatics*, vol. 15, no. 7, pp. 750–757, 2020.

[45] T. Smolarczyk, I. Roterman-Konieczna, and K. Stapor, "Protein secondary structure prediction: a review of progress and directions," *Current Bioinformatics*, vol. 15, no. 2, pp. 90–107, 2020.

[46] Y. Liu, X.-H. Ouyang, Z.-X. Xiao, L. Zhang, and Y. Cao, "A review on the methods of peptide-MHC binding prediction," *Current Bioinformatics*, vol. 15, no. 8, pp. 878–888, 2020.

[47] H. Huang and X. Gong, "A review of protein inter-residue distance prediction," *Current Bioinformatics*, vol. 15, no. 8, pp. 821–830, 2020.

[48] J. Shao and B. Liu, "ProtFold-DFG: protein fold recognition by combining directed fusion graph and PageRank algorithm," *Briefings in Bioinformatics*, vol. 22, no. 3, 2021.

[49] Y. Pang and B. Liu, "SelfAT-fold: protein fold recognition based on residue-based and motif-based self-attention networks," *IEEE/ACM Transactions on Compuational Biology and Bioinformatics*, p. 1, 2020.

[50] H. Wang, Y. Ding, J. Tang, and F. Guo, "Identification of membrane protein types via multivariate information fusion with Hilbert-Schmidt independence criterion," *Neurocomputing*, vol. 383, pp. 257–269, 2020.

[51] Y. Ding, J. Tang, and F. Guo, "Identification of drug-target interactions via dual Laplacian regularized least squares with multiple kernel fusion," *Knowledge-Based Systems*, vol. 204, 2020.

[52] Y. Ding, J. Tang, and F. Guo, "Identification of drug-target interactions via fuzzy bipartite local model," *Neural Computing & Applications*, vol. 32, pp. 10303–10319, 2020.

[53] D. Zhang, H.-D. Chen, H. Zulfiqar et al., "iBLP: an XGBoost-based predictor for identifying bioluminescent proteins," *Computational and Mathematical Methods in Medicine*, vol. 2021, Article ID 6664362, 15 pages, 2021.

[54] D. Wang, Z. Zhang, Y. Jiang et al., "DM3Loc: multi-label mRNA subcellular localization prediction and analysis based on multi-head self-attention mechanism," *Nucleic Acids Research*, vol. 49, no. 8, p. e46, 2021.

[55] F. Y. Dao, H. Lv, D. Zhang, Z. M. Zhang, L. Liu, and H. Lin, "DeepYY1: a deep learning approach to identify YY1-mediated chromatin loops," *Briefings in Bioinformatics*, 2020.