Mathematical Biosciences and Engineering

MBE, 18(6): 8797–8814.
DOI: 10.3934/mbe.2021434
Received: 20 August 2021
Accepted: 30 September 2021
Published: 14 October 2021

http://www.aimspress.com/journal/MBE

Research article

iEnhancer-MFGBDT: Identifying enhancers and their strength by fusing multiple features and gradient boosting decision tree

Yunyun Liang [1,*], Shengli Zhang [2,*], Huijuan Qiao [2] and Yinan Cheng[3]

[1] School of Science, Xi'an Polytechnic University, Xi'an 710048, China
[2] School of Mathematics and Statistics, Xidian University, Xi'an 710071, China
[3] Department of Statistics, University of California at Davis, Davis, CA 95616, USA

*** Correspondence:** Email: yunyunliang88@163.com; shengli0201@163.com.

Abstract: Enhancer is a non-coding DNA fragment that can be bound with proteins to activate transcription of a gene, hence play an important role in regulating gene expression. Enhancer identification is very challenging and more complicated than other genetic factors due to their position variation and free scattering. In addition, it has been proved that genetic variation in enhancers is related to human diseases. Therefore, identification of enhancers and their strength has important biological meaning. In this paper, a novel model named iEnhancer-MFGBDT is developed to identify enhancer and their strength by fusing multiple features and gradient boosting decision tree (GBDT). Multiple features include k-mer and reverse complement k-mer nucleotide composition based on DNA sequence, and second-order moving average, normalized Moreau-Broto auto-cross correlation and Moran auto-cross correlation based on dinucleotide physical structural property matrix. Then we use GBDT to select features and perform classification successively. The accuracies reach 78.67% and 66.04% for identifying enhancers and their strength on the benchmark dataset, respectively. Compared with other models, the results show that our model is useful and effective intelligent tool to identify enhancers and their strength, of which the datasets and source codes are available at https://github.com/shengli0201/ iEnhancer-MFGBDT1.

Keywords: identification; enhancers; multiple features; gradient boosting decision tree

1. Introduction

Enhancers are non-coding DNA fragments, which hold responsibility for regulating gene

expression in both transcription and translation and the production of RNA and proteins [1]. Unlike the proximal elements promoters of the gene, enhancers are distal elements that can be located up to 20kb upstream or downstream away from a gene, or even located on a different chromosome [2]. Such locational variation makes the identification of enhancers challenging. Moreover, genetic variation in enhancers has been demonstrated that it is related to many human illnesses, such as cancer [3,4], disorder [4,5] and inflammatory bowel disease [6]. Genome-wide study of histone modifications has shown that enhancers are a large group of functional elements with many different subgroups, such as strong enhancers and weak enhancers, poised enhancers and inactive enhancers [7]. Because enhancers of different subgroups have different biological activities, understanding enhancers and their subgroups is an important task, especially for the identification of the enhancers and their strength.

Due to the importance of enhancers in genomics and disease, the identification of the enhancers and their strength has become a popular topic in biological research. The pioneering works carried out purely by the experimental techniques include chromatin immunoprecipitation followed by deep sequencing [8–10], DNase I hypersensitivity [11] and genome-wide mapping of histone modifications [12–16]. However, the experimental methods are expensive, time consuming and low accuracy. Therefore, several computational methods were developed in order to fast identify enhancers and their strength in genomes. In 2016, Liu et al. [2] developed a two-layer predictor iEnhancer-2L, which is the first computational model for identifying not only enhancers, but also their strength by pseudo k-tuple nucleotide composition. At the same year, Jia et al. [17] proposed EnhancerPred model by fusing bi-profile Bayes and pseudo-nucleotide composition as multiple features, and a two-step wrapper for feature selection to distinguish between enhancers and non-enhancers and to determine enhancers' strength. In 2018, Liu et al. [18] established the iEnhancer-EL model for identifying enhancers and their strength with ensemble learning approach. In 2019, Nguyen et al. [19] put forward iEnhancer-ECNN model to identify enhancers and their strength using ensembles of convolutional neural networks. At the same year, Tan et al. [20] used ensemble of deep recurrent neural networks for identifying enhancers via dinucleotide physicochemical properties. Le et al. [21] developed iEnhancer-5Step model to identifying enhancers and their strength using hidden information of DNA sequences via Chou's 5-step rule and word embedding. In 2021, Basith et al. [22] proposed Enhancer-IF model by integrative machine learning (ML)-based framework for identifying cell-specific enhancers. At the same year, Cai ea al. [23] established iEnhancer-XG model by using XGBoost as a base classifier and k-spectrum profile, mismatch k-tuple, subsequence profile, position-specific scoring matrix and pseudo dinucleotide composition as feature extraction methods. Le et al. [24] use a transformer architecture based on BERT and 2D convolutional neural network to identify DNA enhancers. Lim et al. [25] proposed iEnhancer-RF model to identify enhancers and their strength by enhanced feature representation using random forest. However, the stability of the model still needs to be improved, especially for identifying the strong enhancers from the weak enhancers.

In this study, we focus on developing a novel model named iEnhancer-MFGBDT to identify enhancers and their strength. Its first layer serves to identify whether a DNA sequence sample is of enhancer or not, while its second layer is to identify whether the identified enhancer as being strong or weak. We fuse k-mer and reverse complement k-mer nucleotide composition based on DNA sequence, and second-order moving average, normalized Moreau-Broto auto-cross correlation and Moran auto-cross correlation based on dinucleotide physical structural property matrix as extracted

multiple features, and a 902-dimensional feature vector is obtained for each enhancer sequence. Then, gradient boosting decision tree (GBDT) algorithm in this study is adopted as the feature selection strategy and also as the classifier. The accuracy of enhancers and their strength on the benchmark dataset with the 10-fold cross-validation are 78.67% and 66.04%, respectively. The accuracy of enhancers and their strength on the independent dataset with the 10-fold cross-validation are 77.50% and 68.50%, respectively. The experimental results indicate that our model improves the accuracies to identify enhancers and their strength, and is a useful supplementary tool.

2. Materials and methods

2.1. Datasets

In order to facilitate comparison, in this study, we adopt the benchmark dataset S constructed by Liu et al. [2], they obtain the 2968 enhancer sequences with 200bp which can be formulated by

$$\begin{aligned} S &= S^{+} \cup S^{-} \\ S^{+} &= S^{+}_{strong} \cup S^{+}_{weak} \end{aligned}, \tag{1}$$

where S^{+} contains 1484 enhancer sequences, S^{-} contains 1484 non-enhancer sequences, S^{+}_{strong} contains 742 strong enhancer sequences, S^{+}_{weak} contains 742 weak enhancer sequence, in which none of the enhancer DNA sequences has the pairwise sequences similarities more than 80%.

2.2. Feature extraction

Suppose that a DNA enhancer sequence D with L nucleic acid residues is expressed by

$$D = B_1 B_2 B_3 \cdots B_i \cdots B_L, \tag{2}$$

$$B_i \in \{A(\text{adenosine}), C(\text{cytidine}), G(\text{guanosine}), T(\text{thymine})\},$$

where B_i denotes the i-th nucleic acid residue of the DNA sequence at the sequence position i. In this study, 902 multiple features are extracted by fusing k-mer nucleotide composition, reverse complementary k-mer, second-order moving average, Moreau-Broto auto-cross correlation, and Moran auto-cross correlation based on dinucleotide property matrix.

2.2.1. K-mer nucleotide composition

K-mer nucleotide composition is a basic feature extraction approach and widely used in different fields of bioinformatics [26–29]. For a enhancer sequence with L nucleotides, the k-mer nucleotide compositions involve all the possible subsequences with length k of the enhancer sequence. We slide along the enhancer sequence with one nucleotide as a step size using a sliding window k. When the subsequence of the enhancer sequence matches with the i-th k-mer

nucleotide composition, the occurrence number of the k-mer is denoted by n_i. f_i represents the occurrence frequency of the i-th k-mer, and can be expressed by

$$f_i = \frac{n_i}{L-k+1}. \tag{3}$$

For each k, we can obtain 4^k k-mer features, here we let $k = 1,2,3$, Finally, each enhancer sequence obtains $4^1 + 4^2 + 4^3 = 84$-dimensional k-mer feature vector.

2.2.2. Reverse complementary k-mer

The reverse complementary k-mer is a variant of the basic k-mer, and abbreviated as RevKmer, in which the k-mers are not expected to be strand-specific, so reverse complements are collapsed into a single feature. For example, when k =2, there are totally 16 basic k-mers, but by removing the reverse complementary k-mers, only 10 different dinucleotides AA, AC, AG, AT, CA, CC, CG, GA, GC and TA are be retained. In other words, we obtain 10 reverse complementary 2-mer features. Let $k = 1,2,3$, 2+10+32 = 44 RevKmer features are extracted, which can be calculated by a web server named Pse-in-One 2.0 [30].

2.2.3. Second-order moving average based on dinucleotide property matrix

As has been reported, DNA physicochemical properties play crucial role in gene expression regulation and genome analysis, and are also closely correlated with the functional non-coding elements [31–33]. In this study, six dinucleotide physical structural properties are adopted, include three the local translational parameters related to shift, slide and rise, and three the local angular parameters related to twist, tilt and roll [34]. The values of six DNA dinucleotide physical structural properties are shown in Table 1. Each DNA physical structural property is normalized for reducing the bias and noise by the following formula

$$\frac{P - P_{\min}}{P_{\max} - P_{\min}}, \tag{4}$$

where P is the original value of the property, $P_{\min}$ and $P_{\max}$ are the minimum and the maximum property values, respectively.

A DNA sequence is a polymer of four nucleotides with A, C, G and T. Any combination of two nucleotides is called dinucleotide. Hence, there are totally 4*4 = 16 basic dinucleotides. First of all each dinucleotide in a DNA sequence is replaced by the value of the physical structural property. Then, each DNA sequence in the datasets can be converted into a matrix $P = (p_{i,j})_{(L-1)\times 6}$, which is named by dinucleotide property matrix (DPM), where L represents the number of nucleic acid

residue in this DNA sequence. $p_{i,j}$ represents the value of the i th dinucleotide corresponding to the j th physical structural property.

Table 1. The original values of the six physical structural properties for the 16 dinucleotides in DNA.

Dinucleotide	Physical structural property					
	Rise	Roll	Shift	Slide	Tilt	Twist
AA/TT	7.65	2.26	1.69	0.026	0.020	0.038
AC/GT	8.93	3.03	1.32	0.036	0.023	0.038
AG/CT	7.08	2.03	1.46	0.031	0.019	0.037
AT	9.07	3.83	1.03	0.033	0.022	0.036
CA/TG	6.38	1.78	1.07	0.016	0.017	0.025
CC/GG	8.04	1.65	1.43	0.026	0.019	0.042
CG	6.23	2.00	1.08	0.014	0.016	0.026
GA/TC	8.56	1.93	1.32	0.025	0.020	0.038
GC	9.53	2.61	1.20	0.025	0.026	0.036
TA	6.23	1.20	0.72	0.017	0.016	0.018

Second-order moving average (SOMA) algorithm is proposed by Alessio et al. [35], which is defined by fusing the idea of the moving average and the second-order difference. SOMA mainly investigate the long-range correlation properties of a stochastic time series.

Let a discrete stochastic time series be $y(i), i = 1, 2, \cdots, L,$ where L is the size of the stochastic series $y(i)$. The algorithm of the SOMA is described as follows

Step 1. Calculate the moving average $\tilde{y}_n(i)$ of the time series $y(i)$ as

$$\tilde{y}_n(i) = \frac{1}{n}\sum_{k=0}^{n-1} y(i-k), \tag{5}$$

where n is the moving average window. When $n \to 0$, then $\tilde{y}_n(i) \to y(i).$

Step 2. For a given moving average window n, $2 \le n < L$, the second-order difference between the $y(i)$ and $\tilde{y}_n(i)$ is defined by

$$\sigma^2_{MA} = \frac{1}{L-n}\sum_{i=n}^{L}[y(i) - \tilde{y}_n(i)]^2, \tag{6}$$

where σ^2_{MA} is a systematic analysis of the properties of $y(i)$ with respect to $\tilde{y}_n(i)$, so σ^2_{MA} is called the second-order moving average.

A dinucleotide property matrix contains 6 columns, each column is considered a time series, in

other words, a dinucleotide property matrix contains 6 time series. Hence, each enhancer DNA sequence is represented by 6 SOMA features for a certain moving average window n. Here, we let $n = 2,3,\cdots,10$, we construct a $6*9=54$ -dimensional SOMA-DPM feature vector for each enhancer sequence.

2.2.4. Moreau-Broto auto-cross correlation based on dinucleotide property matrix

Normalized Moreau-Broto auto-cross correlation (NMBACC) [36] based on dinucleotide property matrix for extracting global sequence information can be described by

$$NMBACC(s,t,\lambda) = \frac{1}{L-\lambda-1}\sum_{i=1}^{L-\lambda-1} p_{i,s} \times p_{i+\lambda,t},\ (s,t = 1,2,\mathrm{L}\ ,6, 0 < \lambda < L) \tag{7}$$

where λ is the lag of the auto-cross correlation along the column in dinucleotide property matrix. $P_{i,s}$ represents the value at the i-th row for the s-th column (s-th property index), $P_{i+\lambda,t}$ represents the value at the $i+\lambda$-th row for the t-th column (t-th property index). When $s=t$, $NMBACC(s,s,\lambda)$ represents the auto- correlation with the same property. When $s \neq t$, $NMBACC(s,t,\lambda)$ represents the cross-correlation with the different property. Here, we let $\lambda = 1,2,3,\cdots,10$, finally, each enhancer sequence obtains a $6*6*10=360$ -dimensional NMBACC-DPM feature vector.

2.2.5. Moran auto-cross correlation based on dinucleotide property matrix

Moran auto-cross correlation (MACC) [37] based on dinucleotide property matrix for extracting global sequence information can be described by

$$MACC(s,t,\lambda) = \frac{\frac{1}{L-\lambda-1}\sum_{i=1}^{L-\lambda-1}\left(p_{i,s}-\overline{p}_s\right)\left(p_{i+\lambda,t}-\overline{p}_t\right)}{\frac{1}{L-1}\sum_{i=1}^{L-1}\left(p_{i,s}-\overline{p}_s\right)\left(p_{i,t}-\overline{p}_t\right)},\ (s,t = 1,2,\mathrm{L}\ ,6, 0 < \lambda < L) \tag{8}$$

where λ is the lag along the column in dinucleotide property matrix, $p_{i,s}$ and $p_{i,t}$ represent the value at the i-th row for the s-th column and t-th column in dinucleotide property matrix, respectively. $p_{i+\lambda,t}$ represents the value at the $(i+\lambda)$-th row for the t-th column in dinucleotide property matrix. $\overline{p}_s$ and $\overline{p}_t$ are the average value for the s-th and t-th column, respectively. When $s=t$, $MACC(s,s,\lambda)$ represents the auto-correlation with the same property. When $s \neq t$,

$MACC(s,t,\lambda)$ represents the cross-correlation with the different property. Here, we let $\lambda = 1,2,3,\cdots,10$, finally, each enhancer sequence obtains a $6*6*10=360$ -dimensional MACC-DPM feature vector.

2.3. Gradient boosting decision tree

Gradient boosting decision tree (GBDT) is a Boosting algorithm based on decision tree as base learner, was proposed by Freidman in 2001 [38, 39]. It builds a decision tree in each iteration to reduce the residual of the current model in the gradient direction. Then linearly combines the decision tree with the current model to obtain a new model. GBDT repeats the iteration until the number of decision trees reaches the specified value, and the final strong learner is obtained. GBDT is commonly used for regression, classification and feature selection. GBDT's advantages include: (a) It flexible processes of various types of data, including both continuous and discrete dataset; (b) It has powerful predictive ability and generalization ability; (c) It has good interpretability and robustness, can automatically discover high-order relationships between features, and does not require data normalization and other processing.

The GBDT classification algorithm process is as follows

Input: training dataset $D=\{(x_1,y_1),(x_2,y_2),\cdots,(x_m,y_m)\}$. Suppose that the maximum iteration number is T, the loss function is $L(y,f(x))$, and m is the number of samples.

(1) Initialize the weak classifier as follows

$$f_0(x)=\arg\min_c \sum_{i=1}^{m} L(y_i,c), \tag{9}$$

c is the constant value that minimizes the loss function, that is, $f_0(x)$ is a tree with only one root node.

(2) For $t=1$ to T

a. For $i=1$ to m, calculate negative gradient as follows

$$r_{ti}=-\left[\frac{\partial L(y_i,f(x_i))}{\partial f(x_i)}\right]_{f(x)=f_{t-1}(x)}=\frac{y_i}{1+\exp(y_i f(x_i))}, \tag{10}$$

where the loss function $L(y,f(x))=\log(1+\exp(yf(x))),\ y\in\{-1,1\}$.

b. Use $L(x_i,r_{ti}),\ i=1,2,\mathrm{L}\ ,m$ to fit a CART regression tree to get the t-th regression tree, and its corresponding leaf node area is R_{tj}, $j=1,2,\mathrm{L}\ ,J$. J is the number of leaf nodes of the regression tree t.

c. For leaf node area $j = 1, 2, \mathrm{L}\ , J$, calculate the best residual fitting value as follows

$$c_{tj} = \arg\min_{c} \sum_{x_i \in R_{tj}}^{m} \log[1 + \exp(-y_i(f_{t-1}(x_i) + c))]. \tag{11}$$

As the above equation is difficult to optimize, c_{tj} is generally replaced by the approximate value as

$$c_{tj} = \frac{\sum_{x_i \in R_{tj}} r_{ti}}{\sum_{x_i \in R_{tj}} |r_{ti}|(|1 - r_{ti}|)}. \tag{12}$$

d. Update the strong classifier by

$$f_t(x) = f_{t-1}(x) + \sum_{j=1}^{J} c_{tj} I(x \in R_{tj}). \tag{13}$$

(3) Get the final strong classifier $f(x)$ by

$$f(x) = f_T(x) = \sum_{t=1}^{J} \sum_{j=1}^{J} c_{tj} I(x \in R_{tj}). \tag{14}$$

Output: $f_T(x)$.

GBDT can not only be used for classification, but also can be used for feature selection by calculating the gini index. The gini index is ranked in descending order by the importance of the feature, the first k features can be selected as needed. In this study, we adopt GBDT to carry out feature selection and classification.

2.4. Cross-validation and performance assessment

In order to save the computational time, 10-fold cross-validation is carried out for each feature to evaluate the identification performance in this study. The dataset is randomly divided into ten subsets with approximately equal size, and the ratio of the testing set to the training set is 1:9. Each subset is in turn taken as a test set and the remaining nine subsets are used to train the GBDT classifier, and finally the average performance measures over the ten validation results are used for performance evaluation. K-fold cross-validation approach can improve the reliability of evaluation, because all of the original data are considered and each subset is tested only once.

To make an objective and comprehensive evaluation, we employ different performance measures [40–43], including sensitivity (Sn), specificity (Sp), accuracy (Acc) and Matthews correlation coefficient (MCC), The MCC value is ranging from -1 to 1, while the values of other three measures range from 0 to 1. They can be formulated as

$$\begin{cases} Sn = 1 - \dfrac{N_-^+}{N^+} & 0 \le Sn \le 1 \\ Sp = 1 - \dfrac{N_+^-}{N^-} & 0 \le Sp \le 1 \\ Acc = 1 - \dfrac{N_-^+ - N_+^-}{N^+ + N^-} & 0 \le Acc \le 1 \\ MCC = \dfrac{1 - \left(\dfrac{N_-^+ - N_+^-}{N^+ + N^-} \right)}{\sqrt{\left(1 + \dfrac{N_+^- - N_-^+}{N^+} \right) \left(1 + \dfrac{N_-^+ - N_+^-}{N^-} \right)}} & -1 \le MCC \le 1 \end{cases} \quad (15)$$

where N^+ represents the total number of the true enhancer sequences investigated, while N_-^+ represents the number of true enhancer sequences incorrectly identified to be non-enhancer sequences; N^- represents the total number of the non-enhancer sequences investigated while N_+^- represents the number of the non-enhancer sequences incorrectly identified to be enhancer sequences.

We also employ the receiver operating characteristic (ROC) curve [44] and the area under the ROC curve (AUC) [45] to evaluate our model. The ROC curve plots the true positive rate (Specificity) as a function of the false positive rate (1-Specificity) for all possible thresholds. The ROC curve is closer to the upper left corner, the better the identification performance is. In other words, the closer the AUC is to 1, the better the identification system is.

3. Results and discussion

3.1. Identification performance on the benchmark dataset

Identifying enhancers is a binary classification problem, which can be divided into two layers, the first layer is devoted to identify whether a DNA sequence is of enhancer or not, while the second layer is committed to identify enhancer sequence as being strong or weak enhancer. In this study, a novel model iEnhancer-MFGBDT is proposed by using multi-features and gradient boosting decision tree. Firstly, the 902 multi-features are extracted for both layers of each enhancer sequence, which contain 84 *k*-mer features, 44 RevKmer features, 54 SOMA-DPM features, 360 NMBACC-DPM features and 360 MACC-DPM features. Next, 156 features for the first layer, 263 features for the second layer are selected from 902 multi-features with the GBDT algorithm by the gini index, respectively. Finally, the GBDT classifier is adopted to implement classification using the 10-fold cross-validation. The Figure 1 shows the operating flow of iEnhancer-MFGBDT model.

Identification results by the 10-fold cross-validation are shown in Table 2 by our iEnhancer-MFGBDT model on the benchmark datastet. From Table 2, we can see that the accuracy reaches 78.67% and 66.04% for the first and second layers on the benchmark dataset, respectively. Meanwhile, the values of Sn, Sp and MCC reach 77.54%, 79.78%, 0.5735 for the first layer, 70.56%, 61.63%, 0.3232 for the second layer. The AUC indicates the probability at which the model ranks a randomly selected positive sample higher than a randomly selected negative sample. In fact, The AUC can measure the overall performance of a given identification system. The ROC curves are

plotted for the both first and second layers, and shown in Figure 2. The AUC values on the benchmark dataset are 0.8615 and 0.7187 for the first layer and the second layer, respectively. Obviously, the second layer is more difficult to identify than the first layer due to their position variation and free scattering.

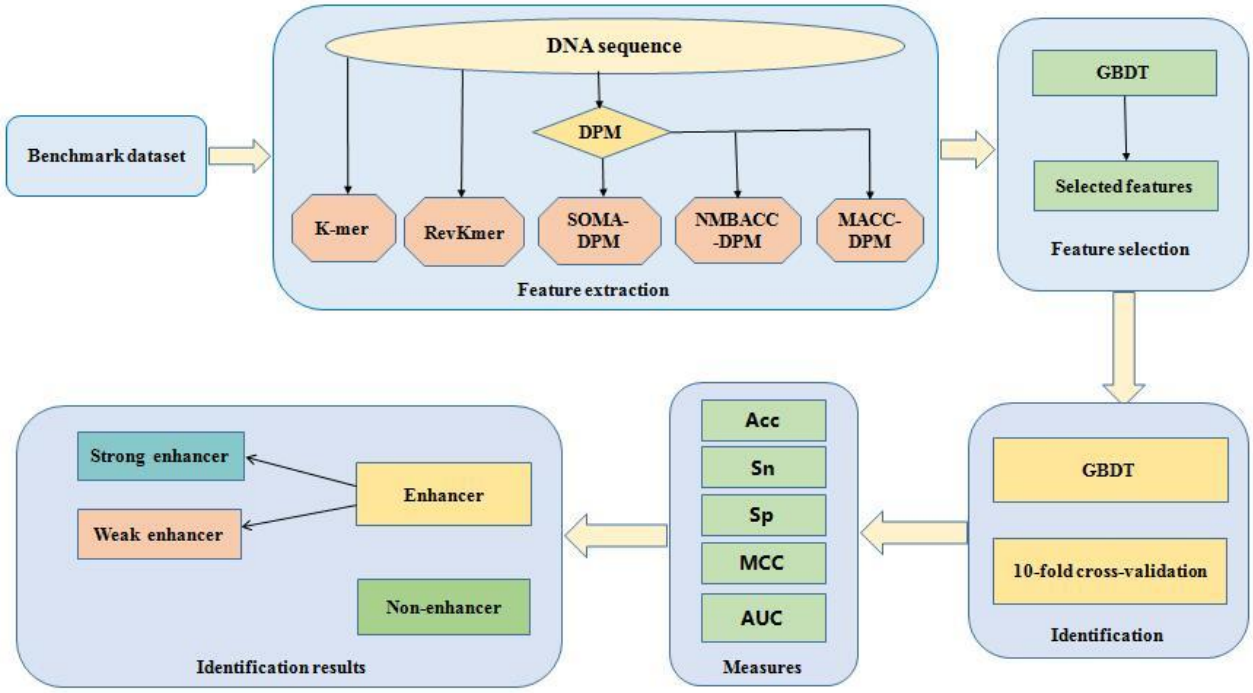

Figure 1. The flowchart of the iEnhancer-MFGBDT model.

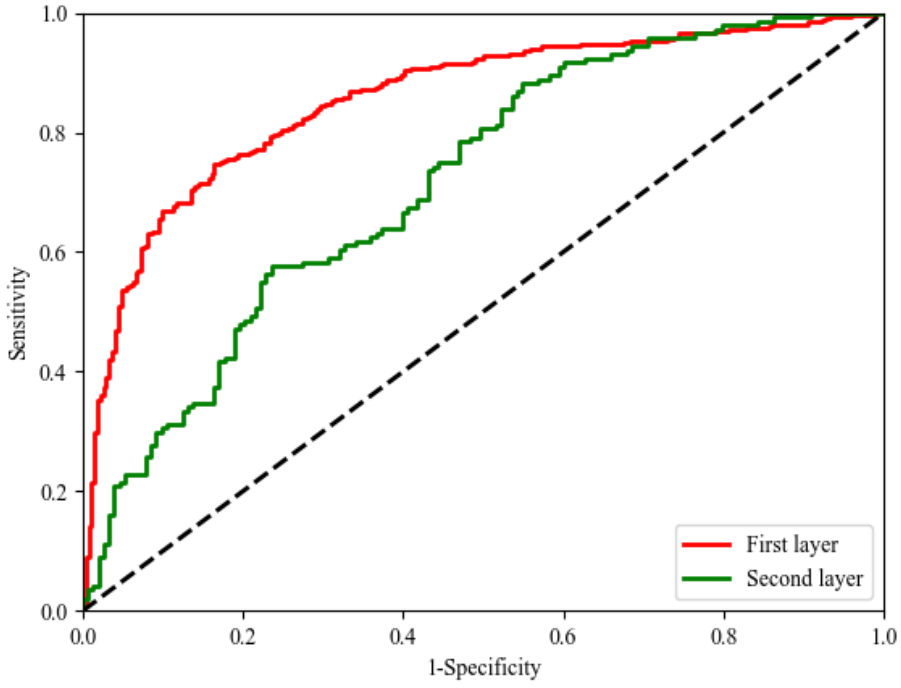

Figure 2. The ROC curves for the first and second layers on the benchmark dataset.

Table 2. The identification performance of iEnhancer-MFGBDT with 10-fold cross validation on the benchmark dataset.

Identification layer	Acc(%)	Sn(%)	Sp(%)	MCC	AUC
First layer	78.67	77.54	79.78	0.5735	0.8615
Second layer	66.04	70.56	61.63	0.3232	0.7187

3.2. Feature group analysis

In this study, we adopt five different approaches to extract features from the benchmark dataset, which are named as K-mer, RevKmer, SOMA-DPM, NMBACC-DPM and MACC-DPM feature group, respectively. For the purpose of obtaining the importance of single feature group, we calculate the performance for K-mer, RevKmer, SOMA-DPM, NMBACC-DPM and MACC-DPM, respectively, and as shown in Table 3. The accuracy of single feature group is lower than that of multiple features after GBDT feature selection (MGBDT) for the both layers. Therefore, the fusion of multiple features is very necessary. From Table 3, we can see that the best identification performance is K-mer, followed by RevKmer, NMBACC-DPM, SOMA-DPM successively, the MACC-DPM is the lowest one for the first layer. Meanwhile, we also can see that the best identification performance is RevKmer, followed by K-mer, SOMA-DPM, MACC-DPM successively, the NMBACC-DPM is the lowest one for the second layer. Among these five feature groups, k-mer and RevKmer are the feature extraction methods based on DNA sequence, SOMA-DPM, NMBACC-DPM and MACC-DPM are the feature extraction methods based on physical structural properties of DNA dinucleotide. Obviously, the DNA sequence-based feature group is superior to physical structural properties-based feature group.

Table 3. Feature group analysis of iEnhancer-MFGBDT with 10-fold cross validation on the benchmark dataset.

Feature group	Acc(%)	Sn(%)	Sp(%)	MCC
First layer				
K-mer	76.99	75.15	78.80	0.5399
RevKmer	76.48	74.08	78.91	0.5364
SOMA-DPM	75.03	73.68	76.30	0.4999
NMBACC-DPM	75.34	74.25	76.43	0.5066
MACC-DPM	74.96	71.33	78.59	0.5000
MGBDT	78.67	77.54	79.78	0.5735
Second layer				
K-mer	61.46	68.36	54.57	0.2319
RevKmer	62.34	69.88	54.64	0.2490
SOMA-DPM	60.65	64.67	56.51	0.2124
NMBACC-DPM	59.77	64.12	55.50	0.1972
MACC-DPM	59.98	60.69	59.37	0.2008
MGBDT	66.04	70.56	61.63	0.3232

3.3. Comparison with feature selection and without feature selection

We construct 902 features by multiple features, and the large dimension will lead to decrease predictive performance, a handicap for the computation and information redundancy. The features selection can help the original classification system achieve a better predictive performance and a lower computational cost by removing redundant features. Hence, finding a suitable dimension reduction method is very important. The gini index is ranked in descending order by importance for GBDT, here, we use "mean" and "gini" as the threshold and criterion for feature selection. Figure 3 shows the accuracy comparison between our model with feature selection and without feature selection. It is obvious that the accuracies have been improved for both layers in the benchmark dataset, and clearly shows that GBDT feature selection method has great effect on improving accuracy. The accuracy is improved by 1.35% and 5.87% for the first layer and the second layer by using GBDT feature selection, respectively. These experimental results show that GBDT is very effective for the benchmark dataset.

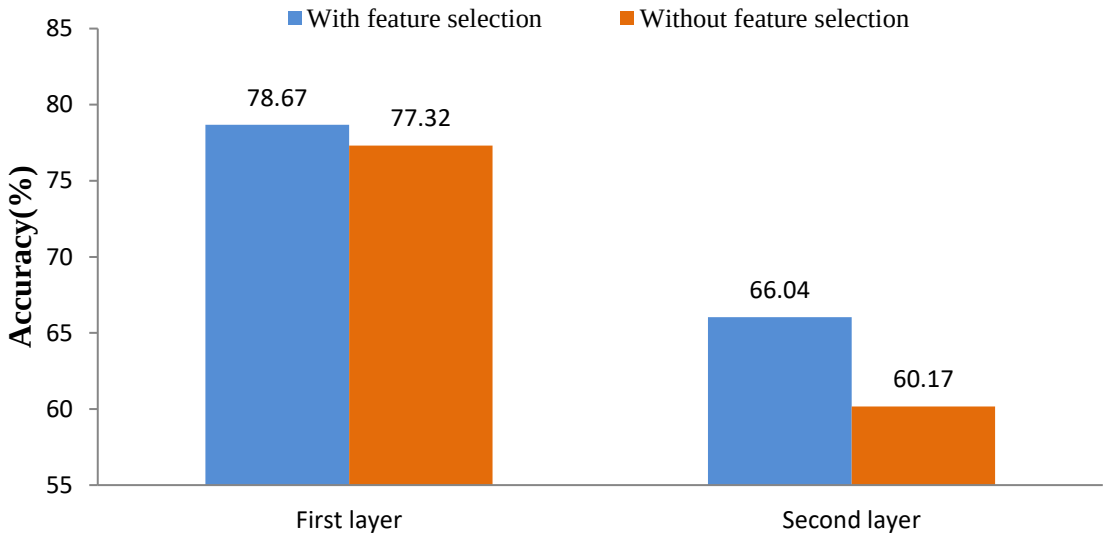

Figure 3. Identification accuracy comparison between with feature selection and without feature selection on the benchmark dataset.

3.4. Comparison with different classifiers

To demonstrate the superiority of GBDT classifier, support vector machine (SVM), extra trees (ET), random forest (RF) and Bagging classifiers are tested successively using the selected features by GBDT based on the 10-fold cross-validation. As shown in Figure 4, the identification accuracy of SVM, ET, RF and Bagging reaches 75.64%, 77.02%, 77.15% and 76.75% for the first layer, and 60.04%, 62.47%, 65.02% and 64.75% for the second layer, respectively. However, the identification accuracy of GBDT reaches 78.67% and 66.04% for the first and second layer, respectively, we can see that from Figure 4, the accuracies of SVM, ET, RF and Bagging are all lower than the accuracies obtained by GBDT for the both layers. The results show that GBDT is more powerful for our benchmark dataset than other classifiers.

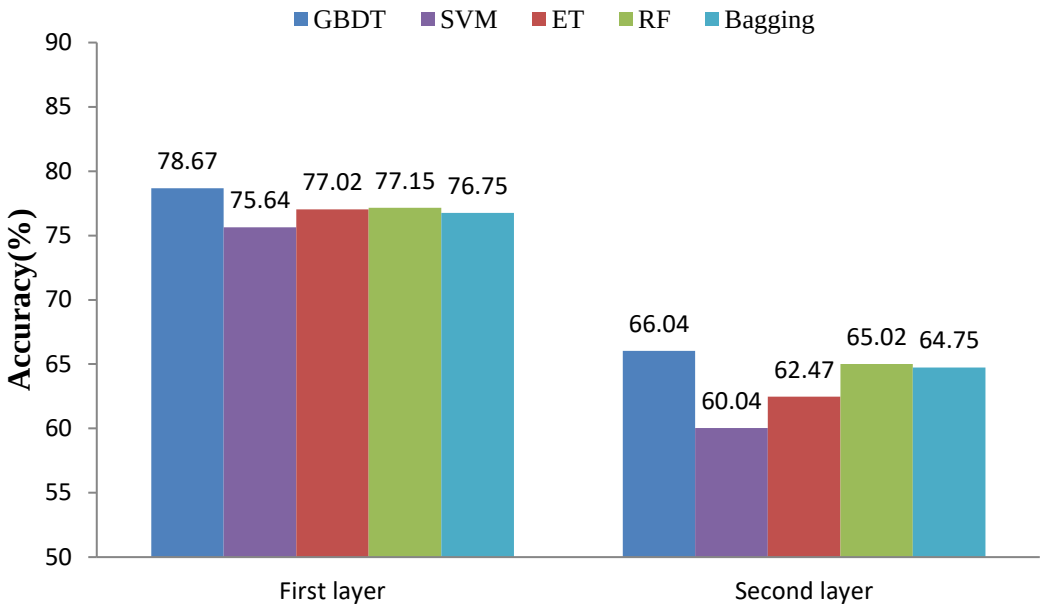

Figure 4. Identification accuracy comparison with different classifiers.

3.5. Independent dataset test

In order to avoid experimental errors, it is persuasive to use an independent dataset to objectively evaluate our model. We adopt the independent dataset also constructed by Liu et al. [2], which contains the 400 enhancer sequences with 200bp, among them, 100 strong enhancer sequences, 100 weak enhancer sequences and 200 non-enhancer sequences, and sequence similarity is less than or equal to 80% . The results obtained by the proposed model using the 10-fold cross-validation on the independent dataset test are given in Table 4. For the first layer, the ACC, Sn, Sp, MCC and AUC reach 77.50%, 76.79%, 79.55%, 0.5607 and 0.8589, respectively. For the second layer, the ACC, Sn, Sp, MCC and AUC reach 68.50, 72.55%, 66.81%, 0.3862 and 0.7524, respectively. The values of these metrics further illustrate the effectiveness of our model.

Table 4. The identification performance of iEnhancer-MFGBDT with 10-fold cross validation on the independent dataset.

Identification layer	Acc(%)	Sn(%)	Sp(%)	MCC	AUC
First layer	77.50	76.79	79.55	0.5607	0.8589
Second layer	68.50	72.55	66.81	0.3862	0.7524

3.6. Performance comparison with other models

The proposed iEnhancer-MFGBDT model, is compared with eight state-of-the-art models: iEnhancer-2L [2], iEnhancerPred [17], iEnhancer-EL [18], iEnhancer-ECNN [19], Tan et al. [20], iEnhancer-XG [23], BERT-2D CNNs [24] and iEnhancer-RF [25]. The values of Acc, Sn, Sp and MCC are listed in Tables 5 and 6.

Table 5. The comparison with other methods in identifying enhancers and their strength on the benchmark dataset.

Model	Acc(%)	Sn(%)	Sp(%)	MCC
First layer				
iEnhancer-2L [2]	76.89	78.09	75.88	0.5400
iEnhancerPred [17]	73.18	72.57	73.79	0.4636
iEnhancer-EL [18]	78.03	75.67	80.39	0.5613
Tan et al.[20]	74.83	73.25	76.42	0.4980
iEnhancer-XG[23]	81.10	75.70	86.50	0.6265
iEnhancer-RF[25]	76.18	73.64	78.71	0.5264
iEnhancer-MFGBDT	78.67	77.54	79.78	0.5735
Second layer				
iEnhancer-2L [2]	61.93	62.21	61.82	0.2400
EnhancerPred [17]	62.06	62.67	61.46	0.2413
iEnhancer-EL [18]	65.03	69.00	61.05	0.3149
Tan et al.[20]	58.96	79.65	38.28	0.1970
iEnhancer-XG[23]	66.74	74.94	58.55	0.3395
iEnhancer-RF[25]	62.53	68.46	56.61	0.2529
iEnhancer-MFGBDT	66.04	70.56	61.63	0.3232

Table 6. The comparison with other methods in identifying enhancers and their strength on the independent dataset.

Model	Acc(%)	Sn(%)	Sp(%)	MCC
First layer				
iEnhancer-2L [2]	73.00	71.00	75.00	0.4604
iEnhancerPred [17]	74.00	73.50	74.50	0.4800
iEnhancer-EL [18]	74.75	71.00	78.50	0.4964
iEnhancer-ECNN [19]	76.90	78.50	75.20	0.5370
Tan et al.[20]	75.50	75.50	76.00	0.5100
iEnhancer-XG[23]	75.75	74.00	77.50	0.5150
BERT-2D CNNs[24]	75.60	80.00	71.20	0.5140
iEnhancer-MFGBDT	77.50	76.79	79.55	0.5607
Second layer				
iEnhancer-2L [2]	60.50	47.00	74.00	0.2180
EnhancerPred [17]	55.00	45.00	65.00	0.1020
iEnhancer-EL [18]	61.00	54.00	68.00	0.2220
iEnhancer-ECNN [19]	67.80	79.10	56.40	0.3680
Tan et al.[20]	68.49	83.15	45.61	0.3120
iEnhancer-XG[23]	63.50	70.00	57.00	0.2720
iEnhancer-MFGBDT	68.50	72.55	66.81	0.3862

For the benchmark dataset, iEnhancer-2L, iEnhancerPred, iEnhancer-EL, Tan et al., iEnhancer-XG and iEnhancer-RF models are adopted for comparison for the both layers, of which

the values of ACC, Sn, Sp and MCC are listed in Table 5. Among the six models, the accuracy for our model is lower than that of iEnhancer-XG model for the both layers, but the stability of our model is higher than that of iEnhancer-XG model. The accuracy for our model is 1.78%, 5.49%, 0.64%, 3.84% and 2.49% higher than the iEnhancer-2L, iEnhancerPred, iEnhancer-EL, Tan et al. and iEnhancer-RF models for the first layer, respectively, and the accuracy for our model is 4.11%, 3.98%, 1.01%, 7.08% and 3.51% higher than the iEnhancer-2L, iEnhancerPred, iEnhancer-EL, Tan et al and iEnhancer-RF models for the second layer, respectively. As shown in Table 5, our model has the best performance and is the most stable model from Sn, Sp and MCC.

For the independent dataset, iEnhancer-2L, iEnhancerPred, iEnhancer-EL and iEnhancer-ECNN, Tan et al., iEnhancer-XG and BERT-2D CNNs models are adopted for comparison for the first layer, of which the values of ACC, Sn, Sp and MCC are listed in Table 6. The accuracy is improved by 0.6%–4.5% for the first layer. From Table 6, we can see that iEnhancer-2L, iEnhancerPred, iEnhancer-EL and iEnhancer-ECNN, Tan et al., and iEnhancer-XG models are adopted for comparison for the second layer, The accuracy is improved by 0.01%-13.5% for the second layer. The test results still show that the performance of iEnhancer-MFGBDT is best on the independent dataset. Our model achieves remarkably better results than other existing models, and make a considerable improvement for performance.

4. Conclusions

In this study, an effective computational tool called enhancers-MFGBDT has been developed for identification of DNA enhancers and their strength. The iEnhancer-MFGBDT model is established by fusing multi-features and GBDT based on the 10-fold cross validation. Compared with the existing models, our model can obtain satisfactory accuracies for the first and second layers on the benchmark dataset and independent dataset. It is anticipated that iEnhancer-MFGBDT will become a very useful high throughput tool for researching enhancers or, at the least, play an important complementary role to the existing models. As pointed out in [46] by Chou and Shen, user-friendly and publicly accessible web-servers represent the future direction for practically developing more useful computational tools, and have increasing impacts on medical science [47]. In the future, we will make great efforts to establish a web-server for the iEnhancer-MFGBDT model to facilitate communication among colleagues in bioinformatics.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (No. 12101480), the Natural Science Basic Research Program of Shaanxi (Nos.2021JM-115, 2021JM-444), and the Fundamental Research Funds for the Central Universities (No. JB210715).

Conflict of interest

The authors declare no conflict of interest.

References

1. N. Omar, W. Y. Shiong, L. Xi, C. C Yee Ling, M. T. D. Abdullah, N. K. Lee, Enhancer prediction in proboscis monkey genome: A comparative study, *J. Telecom. Electron. Computer Eng.*, **9** (2017), 175–179.
2. B. Liu, L. Y. Fang, R. Long, X. Lan, K. C. Chou, iEnhancer-2L: A two-layer predictor for identifying enhancers and their strength by pseudo k-tuple nucleotide composition, *Bioinformatics*, **32** (2016), 362–369.
3. H. M. Herz, Enhancer deregulation in cancer and other diseases, *Bioessays*, **38** (2016), 1003–1015.
4. G. Zhang, J. Shi, S. Zhu, Y. Lan, L. Xu, H. Yuan, et al., DiseaseEnhancer: A resource of human disease-associated enhancer catalog, *Nucleic Acids Res.*, **46** (2018), D78–D84.
5. O. Corradin, P. C. Scacheri, Enhancer variants: Evaluating functions in common disease, *Genome Med.*, 6 (2014), 85.
6. M. Boyd, M. Thodberg, M. Vitezic, J. Bornholdt, K. Vitting-Seerup, Y. Chen, et al., Characterization of the enhancer and promoter landscape of inflammatory bowel disease from human colon biopsies, *Nat. Commun.*, **9** (2018), 1661.
7. D. Shlyueva, G. Stampfel, A. Stark, Transcriptional enhancers: from properties to genome-wide predictions, *Nat. Rev. Genet.*, **15** (2014), 272–286.
8. N. D. Heintzman, B. Ren, Finding distal regulatory elements in the human genome, *Curr. Opin. Genet. Dev.*, **19** (2009), 541–549.
9. N. D. Heintzman, R. K. Stuart, G. Hon, Y. T. Fu, C. W. Ching, R. D. Hawkins, et al., Distinct and predictive chromatin signatures of transcriptional promoters and enhancers in the human genome, *Nat. Genet.*, **39** (2007), 311–318.
10. A. Visel, M. J. Blow, Z. R. Li, T. Zhang, J. A. Akiyama, A. Holt, et al., ChIP-seq accurately predicts tissue-specific activity of enhancers, *Nature*, **457** (2009), 854–858.
11. A. P. Boyle, L. Y. Song, B. K. Lee, D. London, D. Keefe, E. Birney, et al., High-resolution genome-wide in vivo footprinting of diverse transcription factors in human cells, *Genome Res.*, **21** (2011), 456–464.
12. J. Ernst, P. Kheradpour, T. S. Mikkelsen, N. Shoresh, L. D. Ward, C. B. Epstein, et al., Mapping and analysis of chromatin state dynamics in nine human cell types, *Nature*, **473** (2011), 43–49.
13. G. D. Erwin, N. Oksenberg, R. M. Truty, D. Kostka, K. K. Murphy, N. Ahituv, et al., Integrating diverse datasets improves developmental enhancer prediction, *PLoS Comput. Boil.*, **10** (2014), e1003677.
14. M. Feinandez, D. Miranda-Saavedra, Genome-wide enhancer prediction from epigenetic signatures using genetic algorithm-optimized support vector machine, *Nucleic Acids Res.*, **40** (2012), e77.
15. H. A. Firpi, D. Ucar, K. Tan, Discover regulatory DNA elements using chromatin signatures and artificial neural network, *Bioinformatics*, **26** (2010), 1579–1586.
16. N. Rajagopal, W. Xie, Y. Li, U. Wagner, W. Wang, J. Stamatoyannopoulos, et al., RFECS: A random-forest based algorithm for enhancer identification from chromatin state, *PLoS Comput. Boil.*, **9** (2013), e1002968.
17. C. Z. Jia, W. Y. He, EnhancerPred: a predictor for discovering enhancers based on the combination and selection of multiple features, *Sci. Rep.*, **6** (2016) 38741.
18. B. Liu, K. Li, D. S. Huang, K. C. Chou, iEnhancer-EL: Identifying enhancers and their strength with ensemble learning approach, *Bioinformatics*, **34** (2018), 3835–3842.

19. Q. H. Nguyen, T. Nguyen-Vo, N. Q. K. Le, T. T. T. DO, S. Raharja, B. P. Nguyen, iEnhancer-ECNN: Identifying enhancers and their strength using ensemble of convolutional neural networks, *BMC Genom.*, **20** (2019), 951.
20. K. K. Tan, N. Q. K. Le, H. Y. Yeh, M. C. H. Chua, Ensemble of deep recurrent neural networks for identifying enhancers via dinucleotide physicochemical properties, *Cells*, **8** (2019), 767.
21. N. Q. K. Le, E. K. Y. Yapp, Q. T. Ho, N. Nagasundaram, Y. Y. Ou, H. Y. Yeha, iEnhancer-5Step: Identifying enhancers using hidden information of DNA sequences via Chou's 5-step rule and word embedding, *Anal. Biochem.*, **571** (2019), 53–61.
22. S. Basith, M. M. Hasan, G. Lee, L. Y. Wei, B. Manavalan, Integrative machine learning framework for the identification of cell-specific enhancers from the human genome, *Brief. Bioinform.*, (2021), 1–13. doi：10.1093/bib/bbab252.
23. L. J. Cai, X. B. Ren, X. Z. Fu, L. Peng, M. Y. Gao, X. X. Zeng, iEnhancer-XG: Interpretable sequence-based enhancers and their strength predictor, *Bioinformatics*, **37** (2021), 1060–1067.
24. N. Q. K. Le, Q. T. Ho, T. T. D. Nguyen, Y. Y. Ou, A transformer architecture based on BERT and 2D convolutional neural network to identify DNA enhancers from sequence information, *Brief. Bioinform.*, **22** (2021), 1–7.
25. D. Y. Lim, J. Khanal, H. Tayara, K. T. Chong, iEnhancer-RF: Identifying enhancers and their strength by enhanced feature representation using random forest, *Chemometr. Intell. Lab.*, **212** (2021), 104284.
26. W. He, Y. Ju, X. Zeng, X. Liu, Q. Zou, Sc-ncdnapred: A sequence-based predictor for identifying non-coding dna in saccharomyces cerevisiae, *Front. Microbiol.*, **9** (2018), 2174.
27. C. S. Kim, M. D. Winn, V. Sachdeva, K. E. Jordan, K-mer clustering algorithm using a mapreduce framework: application to the parallelization of the inchworm module of trinity, *BMC Bioinform.*, **18** (2017), 467.
28. J. Matias Rodrigues, T. S. Schmidt, J. Tackmann, C. von Mering, Mapseq: Highly efficient k-mer search with confidence estimates, for rRNA sequence analysis, *Bioinformatics*, **33** (2017), 3808–3810.
29. J. S. Wang, S. L. Zhang, PA-PseU: An incremental passive-aggressive based method for identifying RNA pseudouridine sites via Chou's 5-steps rule, *Chemometr. Intell. Lab.*, **210** (2021), 104250.
30. B. Liu, H. Wu, K. C. Chou, An improved package of web servers for generating various modes of pseudo components of DNA, RNA, and protein sequences, *Natural Sci.*, **4** (2017), 67–91.
31. B. Liu, S. Y. Wang, R. Long, K. C. Chou, iRSpot-EL: Identify recombination spots with an ensemble learning approach, *Bioinformatics*, **33** (2017), 35–41.
32. Y. Y. Yao, S. L. Zhang, Y. Y. Liang, iORI-ENST: Identifying origin of replication sites based on elastic net and stacking learning, *SAR QSAR Environ. Res.*, **32** (2021), 317–331.
33. Z. Liu, X. Xiao, D. J. Yu, J. H. Jia, W. R. Qiu, K. C. Chou, pRNAm-PC: Predicting N6-methyladenosine sites in RNA sequences via physical-chemical properties, *Anal. Biochem.*, **497** (2016), 60–67.
34. R. E. Dickerson, Definitions and nomenclature of nucleic acid structure components, *Nucleic Acids Res.*, **17** (1989), 1797–1803.
35. E. Alessio, A. Carbon, G. Castelli, V. Frappietro, Second-order moving average and scaling of stochastic time series, *The European Physical Journal. B: Condensed Matter and Complex Systems*, **27** (2002), 197–200.

36. Y. Y. Liang, S. L. Zhang, Identify Gram-negative bacterial secreted protein types by incorporating different modes of PSSM into Chou's general PseAAC via Kullback–Leibler divergence, *J. Theor. Biol.*, **454** (2018), 22–29.
37. S. L. Zhang, T. Xue, Use Chou's 5 steps rule to identify DNase I hypersensitive sites via dinucleotide property matrix and extreme gradient boosting, *Mol. Genet. Genom.*, **295** (2020), 1431–1442.
38. J. H. Friedman, Greedy Function Approximation: A Gradient Boosting Machine, *Ann. Stat.*, **29** (2001), 1189–1232.
39. N. Alexey, K. Alois, Gradient boosting machines, a tutorial, *Front. Neurorobot.*, **7** (2013), 21.
40. B. Manavalan, S. Basith, T. H. Shin, L. Wei, G. Lee, mAHTPred: A sequence-based meta-predictor for improving the prediction of anti-hypertensive peptides using effective feature representation, *Bioinformatics*, **35** (2019), 2757–2765.
41. J. H. Jia, Z. Liu, X. Xiao, B. X. Liu, K. C. Chou, iPPI-Esml: An ensemble classifier for identifying the interactions of proteins by incorporating their physicochemical properties and wavelet transforms into PseAAC, *J. Theor. Biol.*, **377** (2015), 47–56.
42. B. Liu, K. Li, D. S. Huang, K. C. Chou, iEnhancer-EL: Identifying enhancers and their strength with ensemble learning approach, *Bioinformatics*, **34** (2018), 3835–3842.
43. S. Basith, B. Manavalan, T. H. Shin, G. Lee, iGHBP: Computational identification of growth hormone binding proteins from sequences using extremely randomised tree, *Comput. Struct. Biotec.*, **16** (2018), 412–420.
44. T. Fawcett, An introduction to ROC analysis, *Pattern Recogn. Lett.*, **27** (2006), 861–874.
45. A. P. Bradley, The use of the area under the ROC curve in the evaluation of machine learning algorithms, *Pattern Recogn.*, **30** (1997), 1145–1159.
46. K. C. Chou, H. B. Shen, Review: Recent advances in developing web-servers for predicting protein attributes, *Natural Sci.*, **1** (2009), 63–92.
47. K. C. Chou, Impacts of bioinformatics to medicinal chemistry, *Med. Chem.*, **11** (2015), 218–234.

AIMS Press