Mathematical Biosciences and Engineering

MBE, 20(1): 1037–1057.
DOI: 10.3934/mbe.2023048
Received: 22 July 2022
Revised: 12 September 2022
Accepted: 18 September 2022
Published: 24 October 2022

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

Research article

LangMoDHS: A deep learning language model for predicting DNase I hypersensitive sites in mouse genome

Xingyu Tang[1], Peijie Zheng[1], Yuewu Liu[2], Yuhua Yao[3] and Guohua Huang[1,]*

[1] School of Electrical Engineering, Shaoyang University, Shaoyang 422000, China
[2] College of Information and Intelligence, Hunan Agricultural University, Changsha 410128, China
[3] School of Mathematics and Statistics, Hainan Normal University, Haikou 571158, China

* **Correspondence:** Email: guohuahhn@163.com.

Abstract: DNase I hypersensitive sites (DHSs) are a specific genomic region, which is critical to detect or understand cis-regulatory elements. Although there are many methods developed to detect DHSs, there is a big gap in practice. We presented a deep learning-based language model for predicting DHSs, named LangMoDHS. The LangMoDHS mainly comprised the convolutional neural network (CNN), the bi-directional long short-term memory (Bi-LSTM) and the feed-forward attention. The CNN and the Bi-LSTM were stacked in a parallel manner, which was helpful to accumulate multiple-view representations from primary DNA sequences. We conducted 5-fold cross-validations and independent tests over 14 tissues and 4 developmental stages. The empirical experiments showed that the LangMoDHS is competitive with or slightly better than the iDHS-Deep, which is the latest method for predicting DHSs. The empirical experiments also implied substantial contribution of the CNN, Bi-LSTM, and attention to DHSs prediction. We implemented the LangMoDHS as a user-friendly web server which is accessible at http:/www.biolscience.cn/LangMoDHS/. We used indices related to information entropy to explore the sequence motif of DHSs. The analysis provided a certain insight into the DHSs.

Keywords: DNase I hypersensitive site; genome; CNN; Bi-LSTM; deep learning

1. Introduction

DNase I hypersensitive sites (DHSs) are a specific genomic region in the chromatin, which is of hypersensitivity to cleavage by the DNase I enzyme [1]. DHSs untie its condensed structure, which makes the DNA exposed and accessible to the regulatory proteins. DHSs are functionally associated with the cis-regulatory elements such as promoters, enhancers, suppressors, insulators, as well as locus control regions [2]. Thus, mapping DHSs is becoming one of the most effective methods to precisely identify the location of many different regulatory elements in specific, well-studied genes [3]. Genetic variations in DHSs were found to be implicated in a wide spectrum of common diseases and traits, including Alzheimer's disease [4–8]. For example, DHSs were identified as driver distal regulatory elements in breast cancer, and were responsible for the aberrant expression of neighboring genes [9].

Identifying DHSs is of great interest to cis-regulatory element annotation. With advances in next-generation sequencing, many high-throughput techniques have been developed to detect DHSs in the past decades [5,10–12], such as the Southern blot approach [13] and DNase-seq [14]. Zhang et al. [15] developed a DNase-seq procedure for genome-wide mapping of DHSs in Arabidopsis thaliana and rice, while Wang et al. proposed a modified DNase-seq for genome-wide identification of DHSs in plants [16]. These experimentally verified DHSs were collected to be deposited in several public databases for further exploration [1].

Although these high-throughput techniques have contributed much to the discovery of thousands of DHSs, they have two inherent limitations: they are expensive and laborious, which make them insufficient to complete the challenging task of detecting DHSs from tremendous volumes of genomes. Computational identification is another routine to detect DHSs. The computational identification is defined as computational models or functions which are able to predict DHSs after they are trained by known DHSs. Computational identification is extremely cheaper and faster than high-throughput techniques for DHSs detection, and thus it is becoming an alternative to identify DHSs. Computational identification based on machine learning, especially deep learning, has extensively been applied to predict transcription factor binding sites [17–23] and to mine DNA/RNA motif [24]. For example, Wang et al. [17] created a hybrid convolutional recurrent neural network (RNN) for predicting transcription factor binding sites which obtained the state-of-the-art performance on 66 in vitro datasets. Zhang et al. [18] presented a deep learning-based method for transcription factor-DNA binding signal prediction that was able to deal with up to four transcription factor binding-related tasks. Wang et al. [19] employed fully convolutional neural networks (CNNs), along with gated recurrent units, to localize transcription factor binding sites. Following these successful practices, no less than 10 computational models or methods have been created for DHSs detection over the recent decade. These models or methods can be grouped into traditional machine learning-based methods [25–28], ensemble learning-based methods [29–34], and deep learning-based methods [35,36]. To the best of our knowledge, the first computational predictor for DHSs was the support vector machine-based method, which was proposed by Noble et al. [25] in 2005. This method used nucleotide composition as a representation of DNA sequences. Evidently, the nucleotide composition is unable to sufficiently represent DNA sequences because it drops out information about position and order. Feng et al. [37] used pseudo nucleotide composition [38–40] to integrate local and global sequence-order effects of DNA sequences. The pseudo nucleotide composition is similar to the pseudo amino acid composition which is a popular and effective representation for protein sequences. Liu et al. [30] computed nucleotide composition, reverse nucleotide composition, and pseudo dinucleotide composition to build three respective random forest-based classifiers. Three single random forest-based classifiers were fused as an ensemble classifier for DHSs prediction. Zhang et al. [41] employed reverse complement k-mer and dinucleotide-based auto covariance to represent DNA sequences. Zhang et al. [29] stacked multiple traditional machine learning algorithms to build an ensemble classifier for DHSs prediction. Zhang et al. [29] also employed the LASSO to reduce the dimension of representations and the SMOTE-Tomek to overcome imbalance between positive and negative samples. Zheng et al. [32] extracted composition information and physicochemical properties and used a boosting algorithm to optimize informative representations. Zhang et al. [33] proposed a dinucleotide-based spatial autocorrelation to represent DNA sequences. The aforementioned methods heavily rely on representations since it determines, to a great extent, whether methods are performed well or not. High effective representations are generally difficult to obtain in practice. Deep learning is merging as a representative of next-generation artificial intelligence, exhibiting vast potential to solve challenges unsolved in the past. Deep learning has been applied in a wide range of fields, including academic and industrial communities. Lyu et al. [36] developed a deep learning method for DHSs prediction which employed CNNs, along with the gate mechanism, to extract representation. To deal with variable-length inputs, Lyu et al. [36] used the spatial pyramid pooling [42], which was initially proposed to deal with variable-size images. The CNNs are able to capture local properties, and thus has extensively been used in the fields of image analysis and natural language processing. However, the CNNs are insufficient to capture dependencies between local parts, such as words. In the text sequences, dependencies between words are vital because they determine whether one understands it or not. Dao et al. [43] combined the CNNs and long-short term memory (LSTM), which is a special RNN for DHSs prediction. Dao et al. [43] stacked the CNNs and the LSTM in order, i.e., they first used the CNNs to capture local characteristics, and then used the LSTM to catch dependencies between local characteristics. The CNNs and the LSTM characterized different properties of sequences. Linearly stacking the CNN and the LSTM would lose their respective merits. In this paper, we stacked the CNNs and the LSTM in a parallel manner, which absorbed two respective representations learned by the CNN and the LSTM. In addition, we used feed-forward attention to improve representations by the LSTM.

2. Data

We downloaded DHS datasets from 14 different tissues and 4 developmental stages in mouse which are available at the following website: http://lin-group.cn/server/iDHS-Deep/. These DHSs were collected according to the atlas of the DHSs created by Breeze et al. [44]. Dao et al. [43] further preprocessed these datasets for training of the iDHS-Deep model, including choosing the DHSs of 50 to 300 bp as positive samples, selecting specific DNA fragments as negative samples, removing or reducing the homology between sequences by using CD-Hit [45,46], which is a sequence-clustering tool, and dividing these datasets into the training set and the independent set at the ratio of 7 to 3. The numbers of positive and negative samples were not equal for the Stomach tissues, which were respectively 1062 and 2125 in the training set, and which were respectively 456 and 911 in the independent set. Except for Stomach tissues, the numbers of positive and negative samples were identical for each tissue or each developmental stage. The number of positive samples were respectively 7114, 10,299, 5766, 6519, 7424, 30,929, 6316, 4978, 1612, 2515, 3511, 2877, 1224, 7512, 52,418, 16,172 and 21,247 for 13 tissues (i.e., Forebrain, Midbrain, Hindbrain, Liver, Lung, Heart, Kidney, Limb, Thymus, Craniofacial, Retina, Muller retina, and Neural tube) and 4 developmental stages (i.e., ESC, Early-Fetal, Late-Fetal and Adult) in the training set, while they were respectively 3049, 4414, 2472, 2795, 3183, 13,256, 2708, 2134, 692, 1078, 1506, 1234, 525, 3224, 22,466, 6933, 9106 in the independent set.

3. Methods

As shown in Figure 1, the architecture of the proposed LangMoDHS mainly comprised the embedding, the CNNs, the Bi-LSTM followed by the feed-forward attention, the dropout, the fully-connected layer and the output layer. Unlike the iDHS-Deep [43], the LangMoDHS stacked the CNNs and the Bi-LSTM layer in a parallel manner. DNA sequences were first translated into integer sequences which were actually immediate input to the LangMoDHS. Each character in the DNA sequence was mapped into an integer as follows: A into 1, C into 2, G into 6 and T into 17 [43]. The same character-encoding scheme was adopted by the iDHS-Deep [43]. The integer sequences were embedded into continuous vectors, which were further characterized by the Bi-LSTM, CNNs and feed-forward attention. The output layer consisted of only a neuron that represented the probability of the input sequence containing the DHSs. The LangMoDHS is similar to the Deep6mAPred [47] in terms of the architecture, which is a deep learning method for predicting DNA N6-methyladenosine sites except that the former replaced one-hot encoding with an embedding layer and used two layers of CNN.

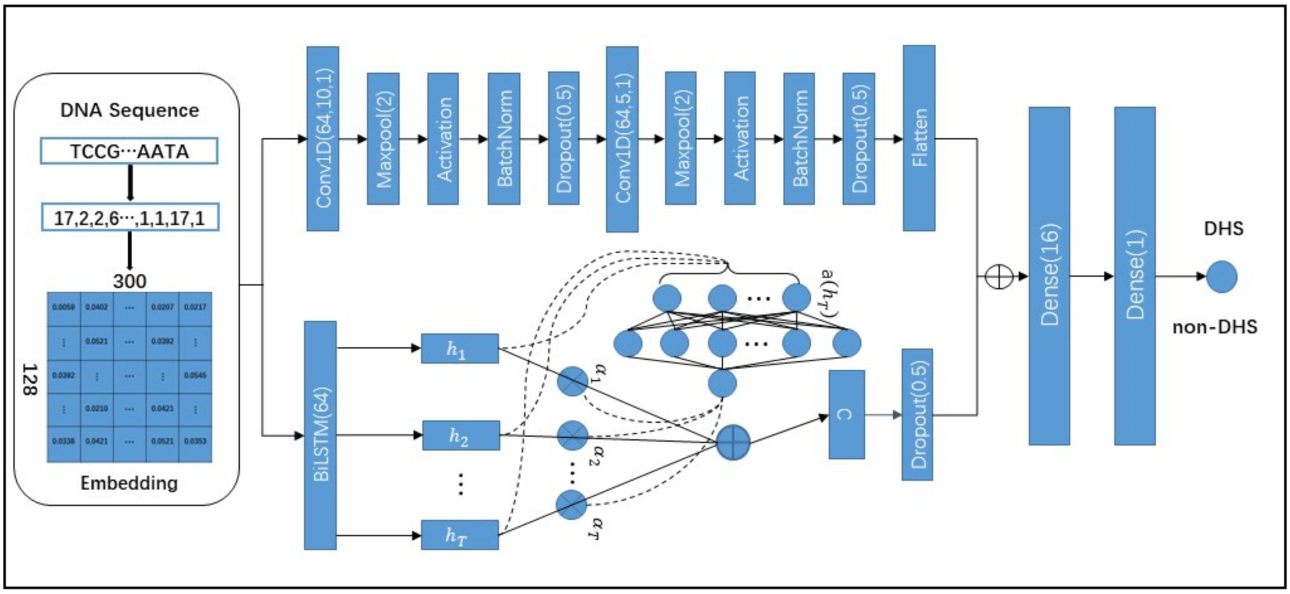


Figure 1. Flowchart of the LangMoDHS. The numbers in parentheses represent the parameters of each layer of the network, h_1, h_2, $\cdots$ h_T represent the vector in the sequence of hidden states, $a(h_T)$ is a learnable function, a_1, a_2, $\cdots$ a_T is the probability vector, the vector c is computed as a weighted average of h_T, $\oplus$ represents element-wise sum and $\otimes$ represents the element-wise product. Dense denotes the fully-connected layer.

3.1. Embedding layer

The embedding layer was intended to bridge the sequences of discrete variables and sequences of continuous vectors. In the deep neural network, the embedding layer is actually a specific neural network analogue to the Word2vec [48,49]. The embedding layer overcame the conventional issues, such as sparsity and no correlation between words. We used embedding in the Keras as the first layer in the LangMoDHS. The Keras is an open-source and extensively used deep learning toolkit.

3.2. CNN

The CNN is increasingly becoming one of the best popular neural networks, which was initially pioneered by Fukushima et al. [50] as an extension of the concept of receptive fields [51]. Since LeCun et al. [52] introduced the gradient back propagation algorithm to train the CNNs, the CNNs have attracted more and more attention, especially from deep learning communities. The CNNs contain two basic operations: convolution and pooling. The convolution is to multiply the input by a fixed convolutional kernel in the same layer. The convolutional kernel is like a filter in the digital signal field, which is shared by the same input. In addition, the CNNs use the pooling to reduce overfitting or computation. Actually, the pooling is a down-sampling method, including average pooling and max pooling. The CNNs are divided into 1D, 2D and 3D CNNs. The 2D CNN and the 3D CNN are generally applied for image data analysis, while the 1D CNN is applied to the field of text analysis. In this study, we used two 1D CNNs, each followed by the max pooling layer.

3.3. Bi-LSTM

LSTM [53] is a special RNN that is different from the CNN. One of the main characteristics of LSTM is to share weights at all the time steps. LSTM is capable of preserving previous semantics by the cell state, which is controlled by the gate mechanism. For example, it uses an input gate to determine how much information is updated, and it uses a forget gate to decide what information is removed in the cell state. Therefore, LSTM is able to capture dependency between words in a sequence and thus is especially suitable to deal with sequence analysis. We used two LSTMs, i.e., Bi-LSTM, to capture the semantic relationship between words.

3.4. Feed-forward attention

The attention mechanism is a newly developed technique of deep learning, and it has extensively been applied in the field of computer vision, natural language processing and bioinformatics. All of the deep learning-based language models, such as transformer and Bert employed attention mechanisms. Even Vaswani et al. declared that attention was all you need [54]. The attention mechanism is actually a scheme to allocate different weights to different parts. In the recent five years, many attention mechanisms have been proposed, including feed-forward attention [55] and self-attention [54]. Here, we used feed-forward attention to compensate for the deficiency of Bi-LSTM. The feed-forward attention is computed by

$$a_t = \frac{exp(e_t)}{\sum_{k=1}^{T} exp(e_k)}, \tag{1}$$

where

$$e_t = \delta(h_t). \tag{2}$$

δ is the learnable parameter and h_t is the hidden state at the time step of t in a Bi-LSTM. The output is a sum of the attentions multiplying corresponding hidden states, which is computed by

$$c = \sum_{t=1}^{T} \theta_t h_t. \tag{3}$$

4. Evaluation metrics

We employed the Receiver Operator Characteristic (ROC) curve, Precision-Recall (PR) curve and F1-score to measure performance. The ROC curve plots the true positive rate (TPR) on the y-axis against the false positive rate (FPR) on the x-axis under the various thresholds. The PR curve plots the Precision on the y-axis against Recall on the x-axis. The F1-score is the summed mean of the Precision and Recall. These metrics are respectively defined by

$$FPR = \frac{F_P}{F_P + T_N}, \tag{4}$$

$$TPR = Recall = \frac{T_P}{T_P + F_N}, \tag{5}$$

$$Precision = \frac{T_p}{T_P + F_P}, \tag{6}$$

$$F1 - score = \frac{2 \times Precision \times Recall}{Precision + Recall}. \tag{7}$$

The more the ROC curve trends to the upper left, the better performance. The more upper right the PR curve trends to upper right, the better performance. The area under the ROC curve (AUROC) and the area under the PR curve (AUPRC) were used to quantitatively assess performance. The AUROC and AUPRC range from 0 to 1, where 1 means a perfect prediction, 0.5 is a random prediction, and 0 is a completely reverse prediction.

5. Results and discussion

We performed a 5-fold cross-validation and an independent test to check the performance of the proposed method. In the 5-fold cross-validation, the training set was randomly divided into 5 parts of equal or approximate size. 4 parts were used to train the model, and the remaining part was used to test the trained model. The process was repeated five times. The independent test was to use the independent set to test the model trained by the training set. Figure 2A,B showed the AUROC values for the 5-fold cross-validations for the training sets from the 14 tissues and the 4 developmental stages, respectively. Obviously, all of the standard deviations of AUROC values over 5-fold cross-validation were less than 0.058, indicating that the LangMoDHS performed stably. Figure 3A,B showed the ROC curves for the independent set. The LangMoDHS achieved the best performance for the Heart tissue (AUROC = 0.960) and performed the worst for the Thymus tissue (AUROC = 0.770). The range of AUROC values for the 14 tissues was 0.19, implying that the LangMoDHS performed differently for different tissues. The LangMoDHS performed stably across the 4 stages. The highest AUROC value was 0.952, the lowest value was 0.910, and the range was 0.042, which was far smaller than that for the 14 tissues. Figures 4 and 5 respectively showed the PR curves and F1-scores for the independent set. The similar phenomenon was observed. For example, the LangMoDHS reached the best AUPRC value, as well as the best F1-score for the Heart tissue, and it achieved the best AUPRC value, as well as the best F1-score for the Early-Fetal stage.

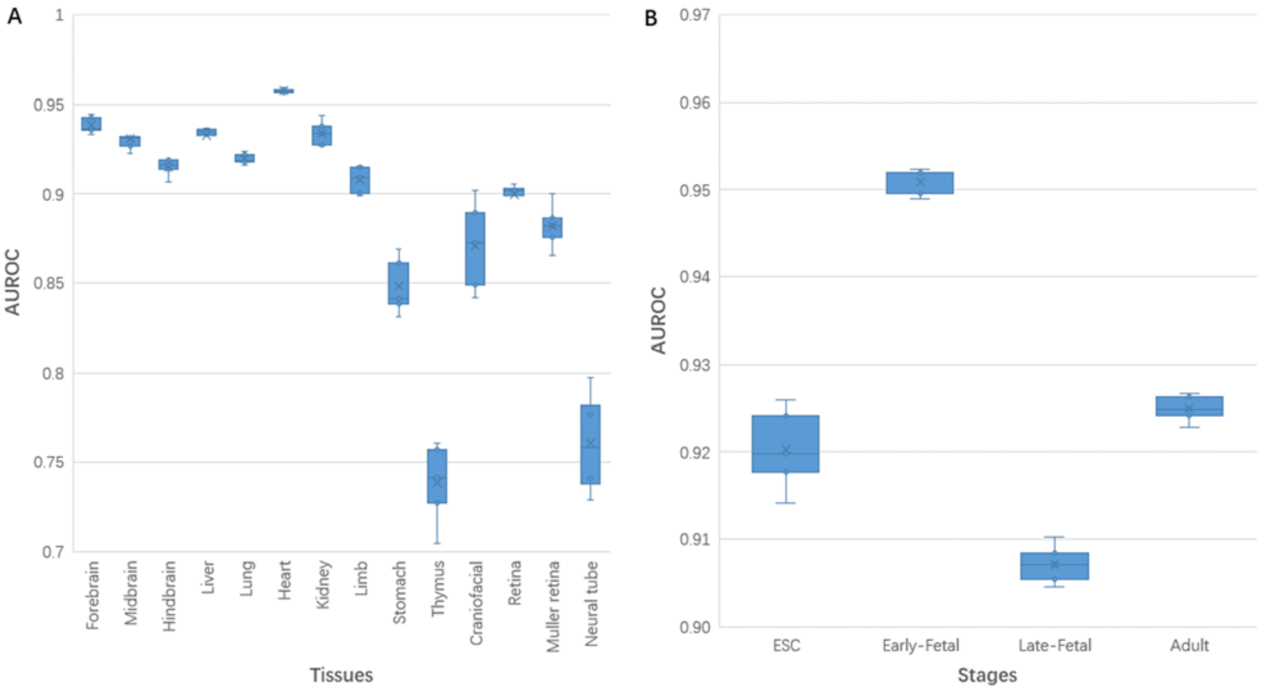


Figure 2. The box plot of AUROC values of 5-fold cross-validations for the training sets from the (A) 14 tissues and (B) from 4 developmental stages.

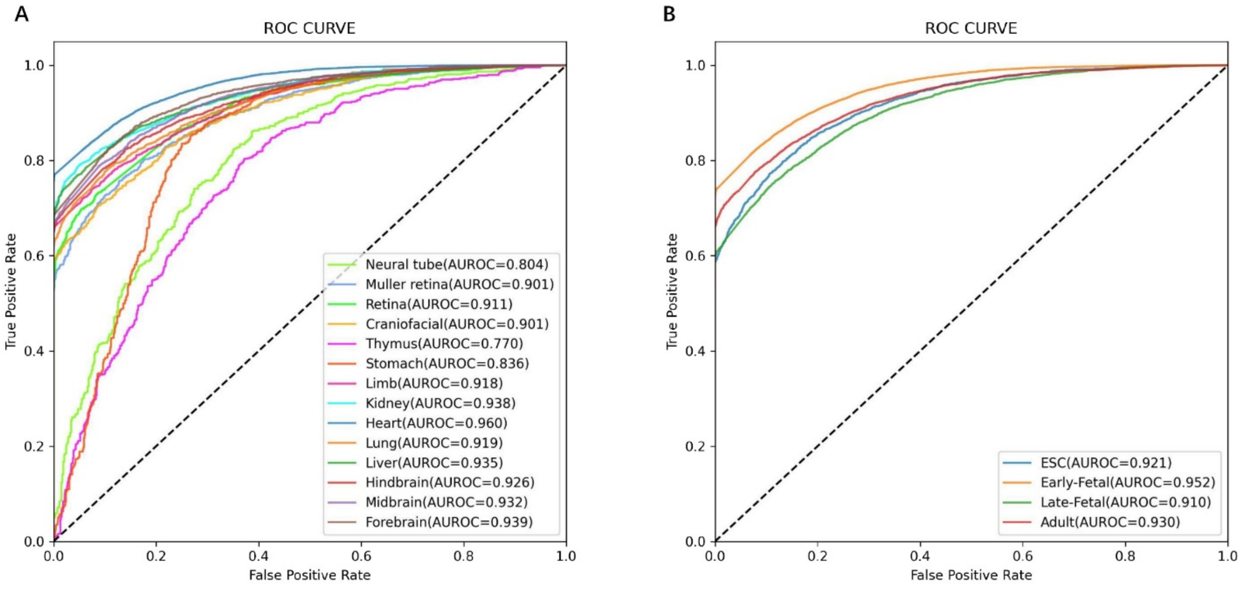


Figure 3. ROC curves of independent tests for the (A) 14 different tissues and (B) 4 different developmental stages.

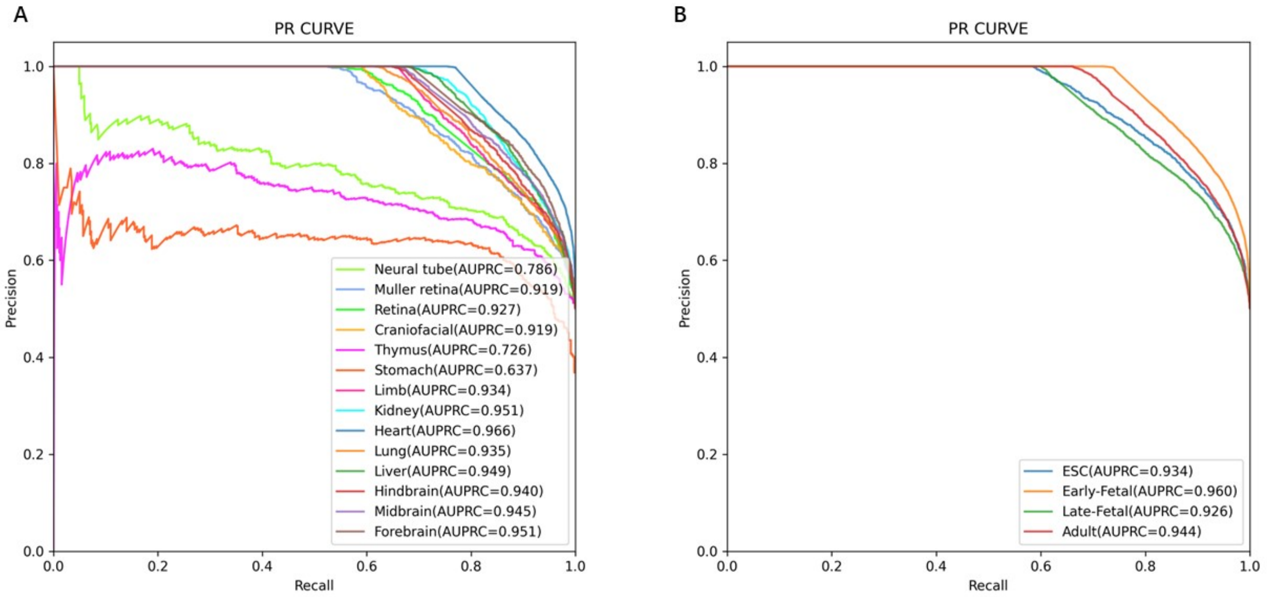


Figure 4. PR curves of independent tests for (A) 14 different tissues and (B) 4 different developmental stages.

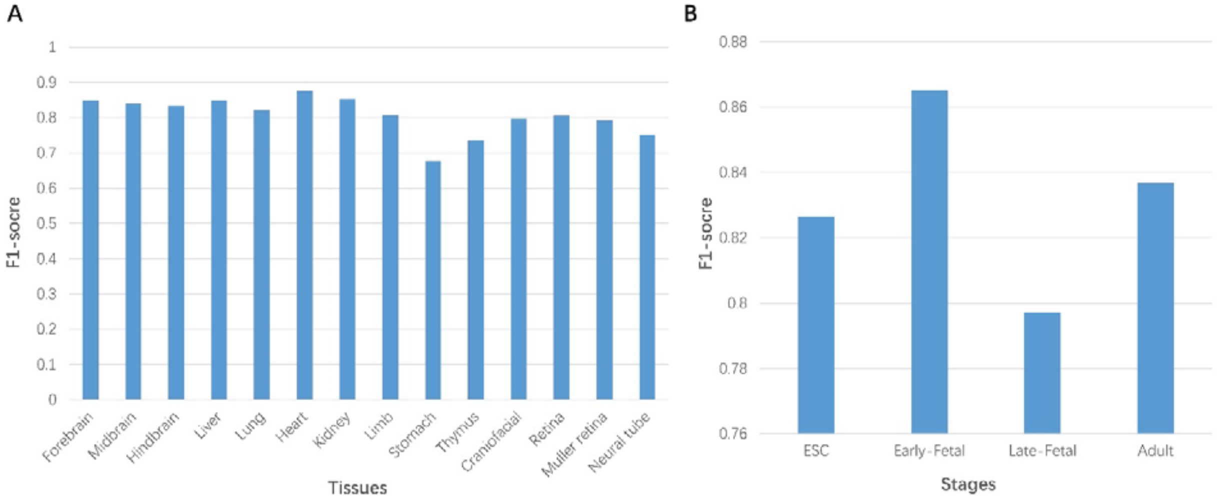


Figure 5. Bar charts of F1-score results for the independent tests for (A) 14 different tissues and (B) 4 different developmental stages.

5.1. Comparison with state-of-the-art method

We compared the LangMoDHS with the iDHS-Deep [43], which is a newly developed method for predicting DHSs. Table 1 listed the AUROC values of 5-fold cross-validation and independent test for the 14 tissues. The iDHS-Deep outperformed the LangMoDHS in both 5-fold cross-validation and independent test over 3 tissues: Stomach, Thymus, and Neural tube, while LangMoDHS completely outperformed iDHS-Deep over 4 tissues: Hindbrain, Liver, Lung and Heart. Over 3 tissues:

Limb, Craniofacial, and Retina, iDHS-Deep performed better in 5-fold cross-validation, while LangMoDHS performed better in independent test. For the Kidney and Midbrain tissues, LangMoDHS was equal to iDHS-Deep in terms of the 5-fold cross-validation, while LangMoDHS performed better than iDHS-Deep in the independent test. Over the Forebrain and the Muller retina tissues, both methods is equivalent in the independent test. Table 2 listed all of the AUROC values in the 5-fold cross-validations and independent tests over 4 developmental stages. Although LangMoDHS performed worse than iDHS-Deep over two developmental stages: ESC and Late-Fetal in the 5-fold cross-validations, the former completely outperformed the latter over all 4 developmental stages in the independent tests.

Table 1. AUROC values of 5-fold cross-validations and independent tests for 14 tissues.

DATASETS	METHOD			
(TISSUES)	IDHS-Deep	LangMoDHS	IDHS-Deep	LangMoDHS
	Training datasets		Independent datasets	
Forebrain	0.934	0.938	0.939	0.939
Midbrain	0.931	0.931	0.920	0.932
Hindbrain	0.911	0.915	0.914	0.926
Liver	0.927	0.932	0.924	0.935
Lung	0.906	0.920	0.885	0.919
Heart	0.955	0.957	0.949	0.960
Kidney	0.934	0.934	0.923	0.938
Limb	0.909	0.907	0.908	0.918
Stomach	0.877	0.848	0.931	0.836
Thymus	0.921	0.738	0.896	0.770
Craniofacial	0.908	0.871	0.894	0.901
Retina	0.902	0.900	0.894	0.911
Muller retina	0.904	0.882	0.901	0.901
Neural tube	0.896	0.763	0.900	0.804

Table 2. AUROC values in the 5-fold cross-validation and independent tests for the 4 developmental stages.

DATASETS	METHOD			
(STAGES)	IDHS-Deep	LangMoDHS	IDHS-Deep	LangMoDHS
	Training datasets		Independent datasets	
ESC	0.923	0.920	0.899	0.921
Early-Fetal	0.949	0.950	0.940	0.952
Late-Fetal	0.923	0.907	0.901	0.910
Adult	0.916	0.925	0.905	0.930

5.2. Tests for cross-tissue and cross-developmental stage evaluation

We further tested LangMoDHS for the ability to predict DHSs across tissues (developmental stages). That is to say, the LangMoDHS trained by the dataset from A tissue (developmental stage), was used to predict DHSs from B tissue (developmental stage). Tables 3 and 4 listed the AUROC value of independent tests across tissues and developmental stages, respectively. Except for seven tissues: Heart, Kidney, Stomach, Thymus, Craniofacial, Muller retina, and Neural tube, LangMoDHS exhibited better performance over other tissues different from itself. For example, the LangMoDHS trained by the Forebrain training set achieved an AUROC value of 0.939 over the independent set from Forebrain, but obtained a better AUROC value (0.950) over the independent set from Heart tissue. This indicated that there existed a potential for LangMoDHS to predict DHSs across tissues. However, not all of the cross-tissue performance of LangMoDHS was better. For example, the LangMoDHS trained by Craniofacial training set reached an AUROC value of 0.901 over the Craniofacial independent set, which was better than those over all of the independent sets from other tissues. The similar phenomenon was observed in Table 4.

Table 3. AUROC values of independent tests across tissues in mouse genomes.

Training datasets Independent datasets	Forebrain	Midbrain	Hindbrain	Liver	Lung	Heart	Kidney	Limb	Stomach	Thymus	Craniofacial	Retina	Muller retina	Neural tube
Forebrain	0.939	0.939	0.922	0.923	0.926	0.944	0.904	0.930	0.789	0.721	0.796	0.914	0.708	0.791
Midbrain	0.923	0.932	0.918	0.917	0.918	0.929	0.901	0.922	0.759	0.725	0.786	0.908	0.730	0.764
Hindbrain	0.913	0.918	0.926	0.910	0.910	0.919	0.895	0.914	0.723	0.698	0.765	0.908	0.734	0.732
Liver	0.901	0.907	0.907	0.935	0.924	0.907	0.922	0.910	0.671	0.658	0.728	0.901	0.747	0.672
Lung	0.885	0.884	0.885	0.896	0.919	0.884	0.892	0.885	0.662	0.655	0.722	0.882	0.732	0.659
Heart	0.950	0.953	0.938	0.938	0.939	0.960	0.920	0.947	0.797	0.710	0.789	0.934	0.700	0.792
Kidney	0.889	0.896	0.901	0.919	0.913	0.884	0.938	0.894	0.594	0.608	0.673	0.903	0.755	0.624
Limb	0.911	0.916	0.910	0.910	0.906	0.920	0.890	0.918	0.744	0.704	0.780	0.898	0.739	0.742
Stomach	0.935	0.945	0.912	0.920	0.920	0.950	0.893	0.917	0.836	0.735	0.812	0.895	0.689	0.801
Thymus	0.899	0.909	0.895	0.902	0.907	0.900	0.883	0.899	0.719	0.770	0.774	0.882	0.743	0.724
Craniofacial	0.892	0.903	0.904	0.897	0.904	0.895	0.890	0.903	0.710	0.712	0.901	0.888	0.754	0.715
Retina	0.888	0.895	0.897	0.899	0.892	0.887	0.894	0.893	0.673	0.674	0.742	0.911	0.747	0.680
Muller retina	0.843	0.855	0.861	0.875	0.866	0.824	0.886	0.855	0.571	0.627	0.680	0.860	0.901	0.613
Neural tube	0.931	0.933	0.913	0.915	0.918	0.933	0.895	0.926	0.750	0.711	0.788	0.910	0.747	0.804

Table 4. AUROC values of independent tests across developmental stages in mouse genomes.

Training datasets	ESC	Early-Fetal	Late-Fetal	Adult
Independent datasets				
ESC	0.921	0.919	0.908	0.911
Early-Fetal	0.940	0.952	0.937	0.934
Late-Fetal	0.890	0.891	0.910	0.903
Adult	0.905	0.902	0.908	0.930

5.3. Ablation test

LangMoDHS consisted mainly of three components: CNN, Bi-LSTM, and feed-forward attention. We investigated further how much the CNN, Bi-LSTM, and feed-forward attention contributed to the recognition of DHSs. Figure 6A–C showed the ROC curves of the independent tests by respectively removing the CNN, Bi-LSTM and attention from LangMoDHS. In contrast, Figure 6D showed the ROC curves of LangMoDHS over the independent tests. It was easy to observe that removing any one of 3 components caused the AUROC values to descend, implying that each contributed substantially to the recognition of DHSs. However, the contributions varied with the tissue (developmental stage) and component. That is to say, some components contributed more for some tissues or developmental stages than for other tissues or stages. For example, the AUROC value after removal of Bi-LSTM was higher than those after respective removal of the CNN and attention for Neural tube tissue, indicating that the CNN and attention contributed more than Bi-LSTM for Neural tube tissue. The AUROC value after removal of attention was more than those after the respective removal of the CNN and Bi-LSTM for Liver tissue, indicating that the CNN and Bi-LSTM contributed more than attention for the Liver tissue. The CNN contributed more than both the Bi-LSTM and attention for 4 developmental stages, the Bi-LSTM contributed more than the attention for 3 developmental stages: ESC, Late-Fetal, and Adult, while attention contributed more than Bi-LSTM for the Early-Fetal developmental stage.

5.4. Sequence motif analysis

We employed information entropy to analyze the motif of DHS sequences. The position-specific nucleotide matrix is defined by

$$Z = \begin{pmatrix} z_{11} & z_{12} & \cdots & z_{1n} \\ z_{21} & z_{22} & \cdots & z_{2n} \\ z_{31} & z_{32} & \cdots & z_{3n} \\ z_{41} & z_{42} & \cdots & z_{4n} \end{pmatrix}, \tag{8}$$

where z_{in} denotes the occurring probability of the nucleotide i at the position j, and n is the length of the sequence. The position-specific nucleotide matrix is estimated by computing the position-specific nucleotide frequencies over all DHS sequences in the benchmark dataset. The nucleotide information entropy and the position information entropy are respectively defined by

$$NP^i = \sum_{j=1}^{n} -Z_{ij}\log{(Z_{ij})} \tag{9}$$

and

$$PP^j = \sum_{i=1}^{4} -Z_{ij}\log{(Z_{ij})}. \tag{10}$$

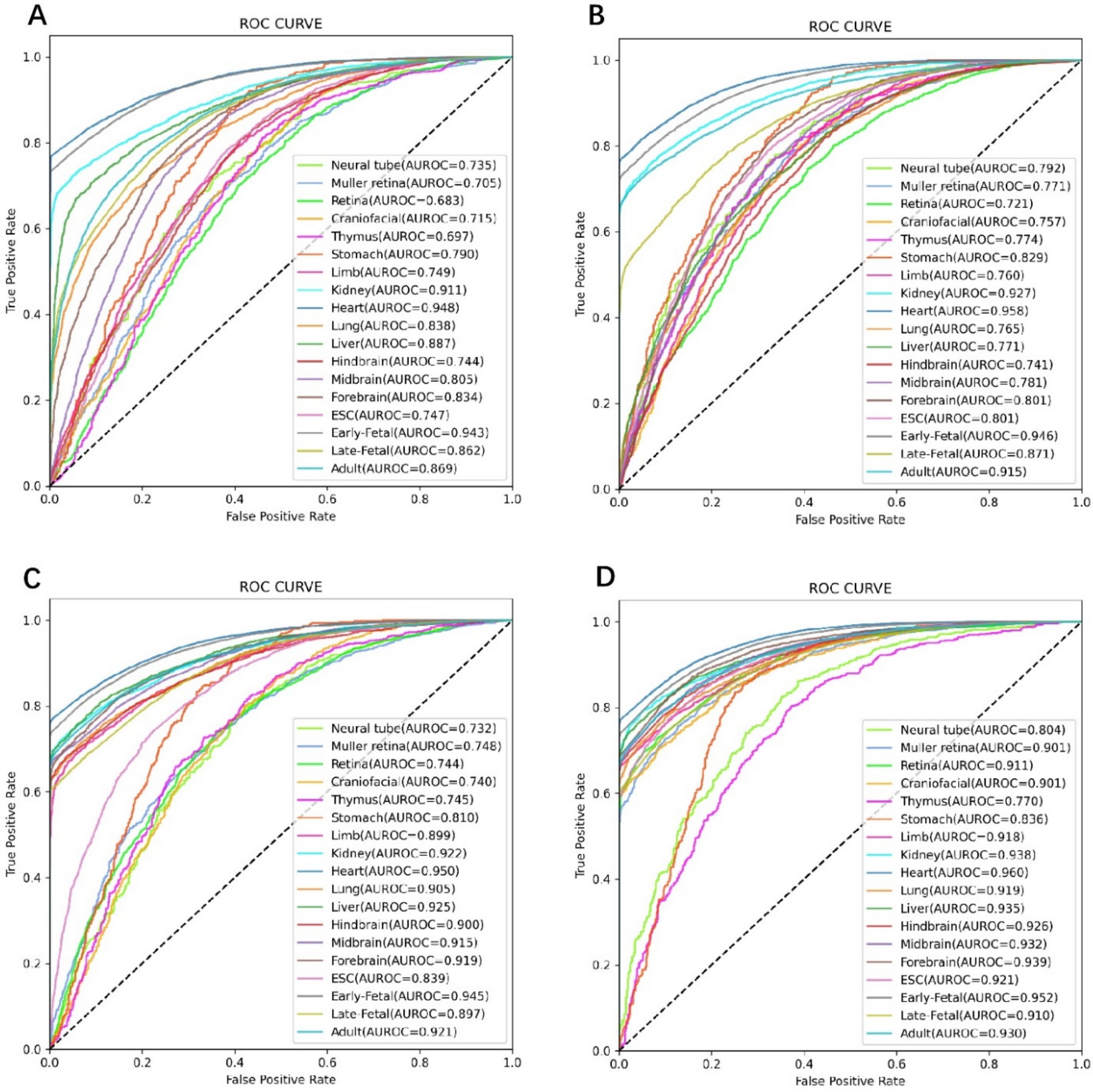


Figure 6. ROC curves of the independent tests by removing the CNN, Bi-LSTM and feed-forward attention. (A) ROC curves of only Bi-LSTM. (B) ROC curves of Bi-LSTM + feed-forward attention. (C) ROC curves of CNN + BiLSTM. (D) ROC curves of the LangMoDHS.

Table 5. Nucleotide information entropy in different tissues of the mouse genome.

DATASETS	Information entropy											
(TISSUES)	A			T			C			G		
	POS	NEG	ALL	POS	NEG	ALL	POS	NEG	ALL	POS	NEG	ALL
Forebrain	7.782	7.997	7.922	7.793	8.001	7.914	7.782	7.999	7.909	7.789	7.997	7.925
Midbrain	7.729	7.998	7.912	7.736	8.001	7.901	7.728	7.997	7.895	7.732	8.000	7.913
Hindbrain	7.796	7.997	7.927	7.804	7.994	7.918	7.794	7.997	7.917	7.795	7.994	7.923
Liver	7.775	8.001	7.918	7.769	7.998	7.908	7.758	7.998	7.904	7.768	8.000	7.915
Lung	7.770	7.993	7.910	7.761	7.993	7.902	7.767	7.993	7.904	7.763	7.990	7.908
Heart	7.842	7.992	7.935	7.837	7.994	7.924	7.831	7.992	7.919	7.844	7.993	7.936
Kidney	7.732	7.990	7.895	7.724	7.994	7.894	7.725	7.988	7.890	7.727	7.988	7.891
Limb	7.781	8.002	7.926	7.787	8.001	7.916	7.777	7.996	7.911	7.786	7.999	7.925
Stomach	7.788	7.999	7.957	7.767	7.992	7.934	7.765	7.996	7.937	7.772	8.000	7.956
Thymus	7.592	7.979	7.867	7.613	7.980	7.857	7.602	7.984	7.859	7.612	7.987	7.881
Craniofacial	7.620	7.994	7.880	7.634	7.992	7.870	7.632	7.994	7.871	7.626	7.991	7.878
Retina	7.714	7.984	7.891	7.705	7.986	7.882	7.718	7.986	7.886	7.717	7.982	7.891
Muller retina	7.571	7.990	7.862	7.560	7.995	7.873	7.581	7.994	7.876	7.555	7.988	7.858
Neural tube	7.770	8.001	7.908	7.695	7.997	7.890	7.694	7.994	7.888	7.690	7.994	7.901

Table 6. Nucleotide information entropy in different developmental stages of the mouse genome.

DATASETS	Information Entropy											
	A			T			C			G		
(STAGES)	POS	NEG	ALL	POS	NEG	ALL	POS	NEG	ALL	POS	NEG	ALL
ESC	7.717	7.992	7.904	7.708	8.000	7.888	7.710	7.993	7.883	7.720	7.997	7.908
Early-Fetal	7.830	7.995	7.934	7.828	7.998	7.924	7.821	7.996	7.920	7.832	7.996	7.935
Late-Fetal	7.650	7.998	7.888	7.656	8.000	7.885	7.652	7.998	7.881	7.659	7.998	7.890
Adult	7.763	7.992	7.908	7.754	7.994	7.900	7.757	7.993	7.900	7.755	7.992	7.905

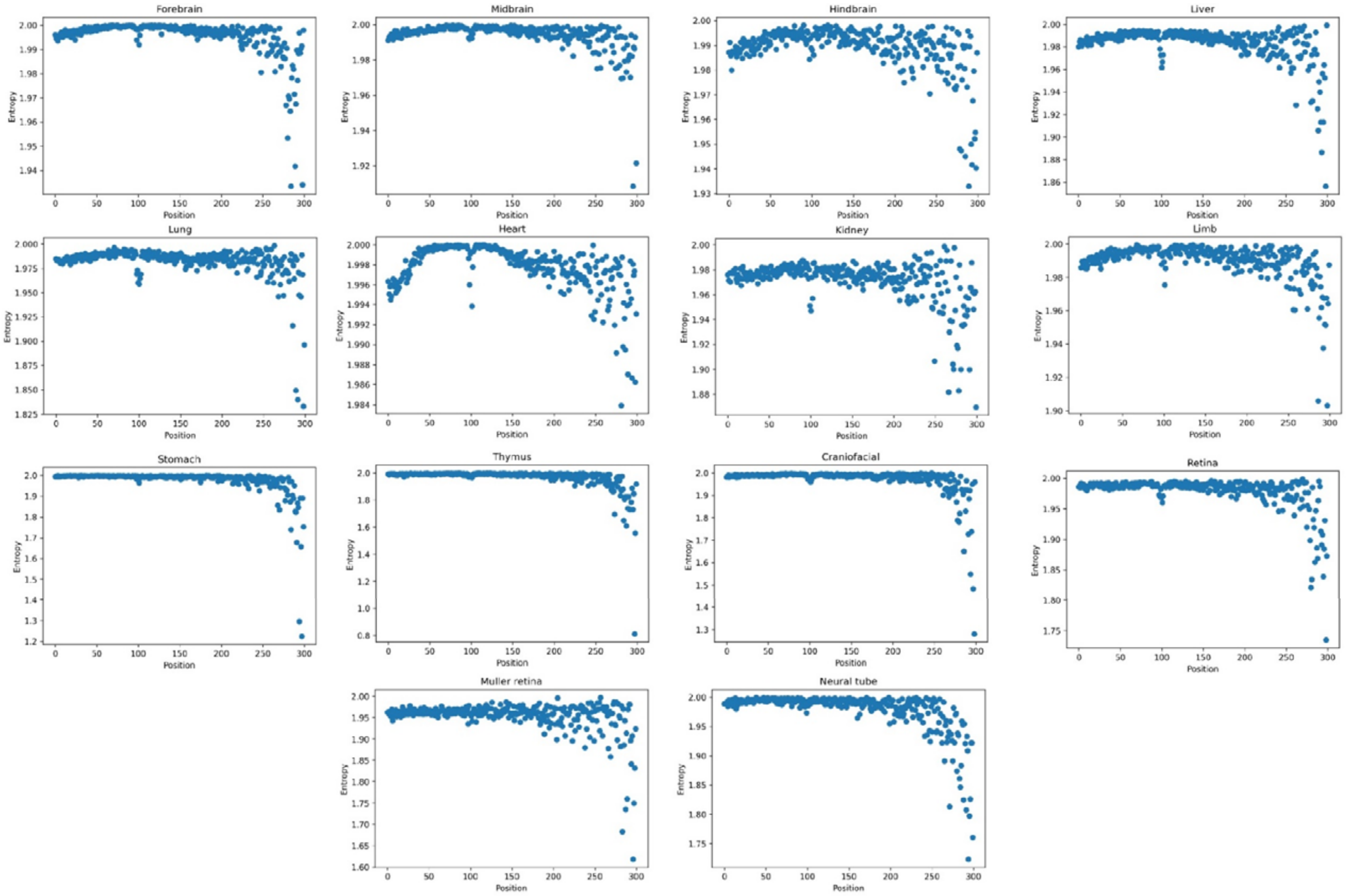


Figure 7. Position information entropy for all tissues.

The lower the information entropy, the more certain the distribution of the nucleotides in the sequences. Tables 5 and 6 showed the nucleotide information entropy for all 14 the tissues and 4 developmental stages, respectively. The nucleotide information entropy of DHSs was lower than those of non-DHSs, indicating that the distribution of nucleotides of DHSs was more certain than those of non-DHSs. The nucleotide information entropy was tissue-specific and stage-specific. For example, the nucleotide information entropy for the Muller retina tissue was lower than those for the Kidney tissue, and the nucleotide information entropy for the Early-Fetal stage was lower than those for the Late-Fetal stage. This implied that the distribution of nucleotides was tissue-specific and stage-specific. Figures 7 and 8 showed the positions of entropy values for all 14 the tissues and all 4 stages, respectively. It can be obviously observed that the position information entropy in the range of 230 to 300 was far less than 2, implying that the distribution of nucleotides in these regions were not completely stochastic. In addition, at the nearby position of 100, the position information entropy for tissues was lower than 2, indicating that the distribution of nucleotides was more certain in these regions.

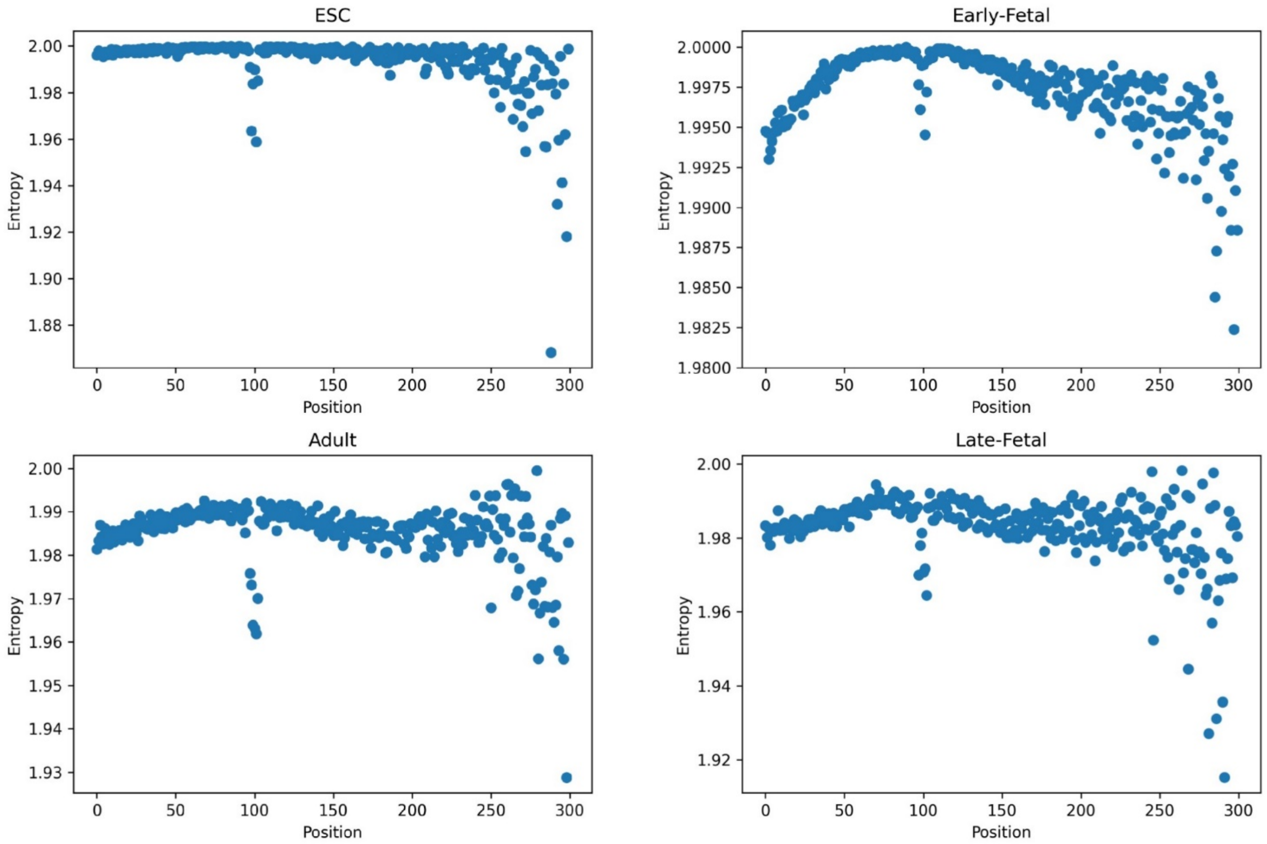
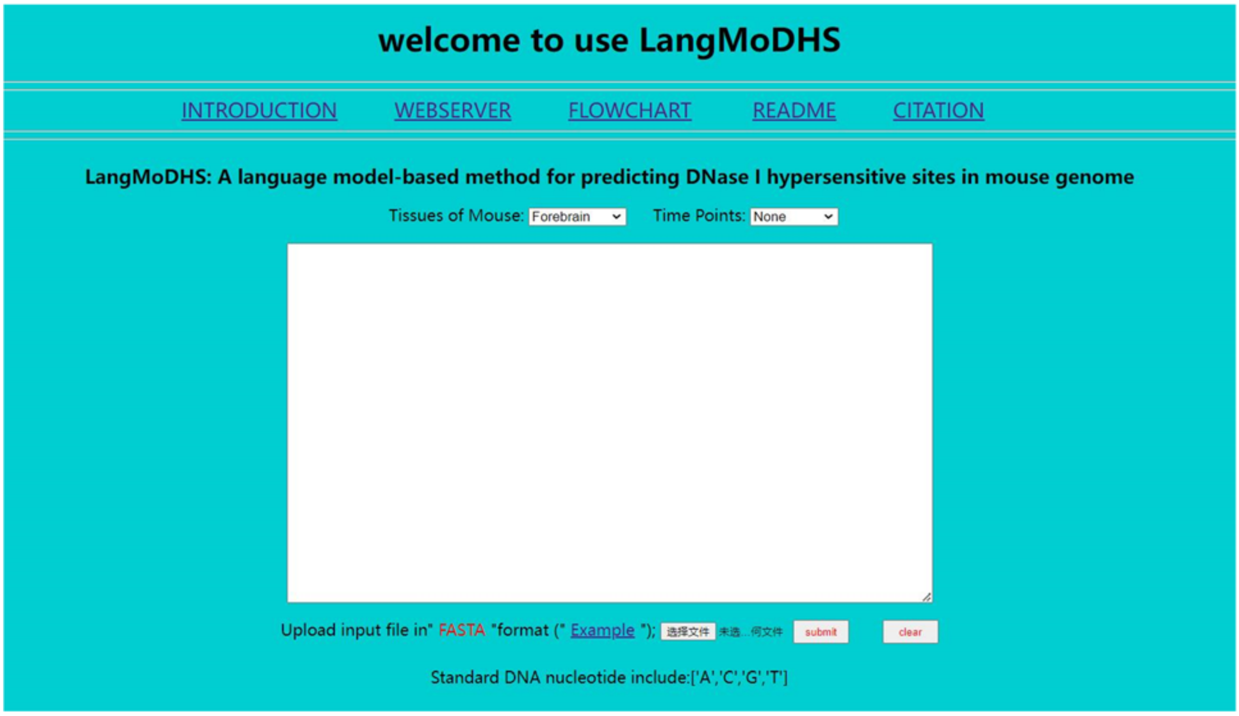


Figure 8. Position information entropy for all developmental stages.

5.5. Web server

To facilitate better using DHS sites and non-DHS sites, we have provided a useful web server at http:/www.biolscience.cn/LangMoDHS/. The web server interface is illustrated in Figure 9. Users first select the organization or developmental stage that needs to be predicted. Then, users input the sequence into the inputting box in the FASTA format. Alternatively, users upload a sequence file in the

FASTA format. Finally, by clicking the submit button, users will get the predictive result in a certain amount of time which is determined by the number of inputted sequences.


welcome to use LangMoDHS

LangMoDHS: A language model-based method for predicting DNase I hypersensitive sites in mouse genome

Tissues of Mouse: Forebrain   Time Points: None

Upload input file in" FASTA "format (" Example "); 选择文件 未选...何文件

Standard DNA nucleotide include:['A','C','G','T']

Figure 9. Web server page of the LangMoDHS.

5.6. Discussion

DHSs play a key role in the cellular process. Sequence motif of DHSs is complicated, and thus, identifying DHSs is a challenging task at present. We have presented a deep language model for detecting DHSs in mouse genome. Extensive experiments showed that the LangMoDHS is an effective and efficient method for detecting DHSs. However, LangMoDHS performed differently with tissues and developmental stages. The LangMoDHS performed best for the Heart tissue, where the AUROC, AUPRC, and F1-score values were 0.960, 0.966, and 0.875, respectively, while it performed worse for 2 tissues, i.e., Thymus and Stomach, where the minimum AUROC and the minimum AUPRC values were 0.770 and 0.637, respectively. The range between the maximum and the minimum AUPRC value was up to 0.329, indicating that the sequence motif of DHSs would vary with tissue. This analysis also indicated that LangMoDHS is tissue-specific and stage-specific to predict DHSs.

It is desirable to develop a universal method which is able to detect DHSs in all tissues or species. However, there is a difference between tissues and species, so it is very difficult to develop such a universal method in practice. Like the iDHS-Deep [43], the LangMoDHS exhibited a certain ability to detect DHSs across tissues or developmental stages. The LangMoDHS achieved better or competitive performance across tissues and developmental stages, indicating that these tissues and stages would be of a similar mechanism of DHSs.

As mentioned previously, there are many computational approaches to detect DHSs. Compared with the methods [25–31,33–35,37], the LangMoDHS is an end-to-end method which requires no artificial design of features. The iDHS-Deep [43] is a newly developed deep-learning-based method to predict DHSs. The iDHS-Deep consisted mainly of two CNN layers and LSTM. The LSTM was attached at the end of the second CNN layer. The main difference between the iDHS-Deep and the LangMoDHS is that the latter used CNNs and Bi-LSTM in a parallel manner. The CNNs and the Bi-LSTM capture different characterization of the DHSs sequences respectively. Therefore, using CNNs and Bi-LSTM in a parallel manner would be more helpful to accumulate different characterization than stacking CNNs and Bi-LSTM in order. This might be a reason why the LangMoDHS performed better than the iDHS-Deep for most tissues and stages. Another difference is that the LangMoDHS to use feed-forward attention to improve the representations captured by the Bi-LSTM. Although the LangMoDHS exhibited competitive performance, the interpretability needs improving.

6. Conclusions

Due to limitations of the methods or techniques, precisely and high effectively identifying DHSs remains challenging. We have presented a deep learning-based language model for computationally predicting DHSs in mouse genomes. Our method achieved competitive performance with the state-of-the-art methods. We developed a user-friendly web server to facilitate the identification of DHSs. The LangMoDHS presented has certain ability to predict DHSs across tissues and across stages, and it is tissue-specific and stage-specific. The nucleotide distributions of DHSs in some regions, such as nearby the position of 100 and the range from 230 to 300, is more certain.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (62272310, 62162025), Hunan Provincial Natural Science Foundation of China (2022JJ50177, 2020JJ4034), Scientific Research Fund of Hunan Provincial Education Department (21A0466, 19A215), and Shaoyang University Innovation Foundation for Postgraduate (CX2022SY046).

Conflict of interest

The authors declare there is no conflict of interest.

References

1. T. Zhang, A. P. Marand, J. Jiang, PlantDHS: A database for DNase I hypersensitive sites in plants, *Nucleic. Acids. Res.*, **44** (2016), D1148–D1153. https://doi.org/10.1093/nar/gkv962
2. D. S. Gross, W. T. Garrard, Nuclease hypersensitive sites in chromatin, *Annu. Rev. Biochem.*, **57** (1988), 159–197. https://doi.org/10.1146/annurev.bi.57.070188.001111
3. G. E. Crawford, I. E. Holt, J. C. Mullikin, D. Tai, E. D. Green, T. G. Wolfsberg, et al., Identifying gene regulatory elements by genome-wide recovery of DNase hypersensitive sites, *Proc. Natl. Acad. Sci.*, **101** (2004), 992–997. https://doi.org/10.1073/pnas.0307540100
4. M. M. Carrasquillo, M. Allen, J. D. Burgess, X. Wang, S. L. Strickland, S. Aryal, et al., A candidate regulatory variant at the TREM gene cluster associates with decreased Alzheimer's disease risk and increased TREML1 and TREM2 brain gene expression, *Alzheimer's Dementia*, **13** (2017), 663–673. https://doi.org/10.1016/j.jalz.2016.10.005

5. W. Meuleman, A. Muratov, E. Rynes, J. Halow, K. Lee, D. Bates, et al., Index and biological spectrum of human DNase I hypersensitive sites, *Nature*, **584** (2020), 244–251. https://doi.org/10.1038/s41586-020-2559-3
6. M. T. Maurano, R. Humbert, E. Rynes, R. E. Thurman, E. Haugen, H. Wang, et al., Systematic localization of common disease-associated variation in regulatory DNA, *Science*, **337** (2012), 1190–1195. https://doi.org/10.1126/science.1222794
7. 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. https://doi.org/10.1038/nature09906
8. M. Mokry, M. Harakalova, F. W. Asselbergs, P. I. de Bakker, E. E. Nieuwenhuis, Extensive association of common disease variants with regulatory sequence, *PLoS One*, **11** (2016), e0165893. https://doi.org/10.1371/journal.pone.0165893
9. D. Weghorn, F. Coulet, K. M. Olson, C. DeBoever, F. Drees, A. Arias, et al., Identifying DNase I hypersensitive sites as driver distal regulatory elements in breast cancer, *Nat. Commun.*, **8** (2017), 1–16. https://doi.org/10.1038/s41467-017-00100-x
10. W. Jin, Q. Tang, M. Wan, K. Cui, Y. Zhang, G. Ren, et al., Genome-wide detection of DNase I hypersensitive sites in single cells and FFPE tissue samples, *Nature*, **528** (2015), 142–146. https://doi.org/10.1038/nature15740
11. G. E. Crawford, S. Davis, P. C. Scacheri, G. Renaud, M. J. Halawi, M. R. Erdos, et al., DNase-chip: A high-resolution method to identify DNase I hypersensitive sites using tiled microarrays, *Nat. Methods*, **3** (2006), 503–509. https://doi.org/10.1038/nmeth888
12. J. Cooper, Y. Ding, J. Song, K. Zhao, Genome-wide mapping of DNase I hypersensitive sites in rare cell populations using single-cell DNase sequencing, *Nat. Protoc.*, **12** (2017), 2342–2354. https://doi.org/10.1038/nprot.2017.099
13. G. E. Crawford, I. E. Holt, J. Whittle, B. D. Webb, D. Tai, S. Davis, et al., Genome-wide mapping of DNase hypersensitive sites using massively parallel signature sequencing (MPSS), *Genome Res.*, **16** (2006), 123–131. https://doi.org/10.1101/gr.4074106
14. L. Song, G. E. Crawford, DNase-seq: A high-resolution technique for mapping active gene regulatory elements across the genome from mammalian cells, *Cold Spring Harbor Protoc.*, **2010** (2010), pdb.prot5384. https://doi.org/10.1101/pdb.prot5384
15. W. Zhang, J. Jiang, Genome-wide mapping of DNase I hypersensitive sites in plants, in *Plant Functional Genomics*, Humana Press, **1284** (2015), 71–89. https://doi.org/10.1007/978-1-4939-2444-8_4
16. Y. Wang, K. Wang, Genome-wide identification of DNase I hypersensitive sites in plants, *Curr. Protoc.*, **1** (2021), e148. https://doi.org/10.1002/cpz1.148
17. S. Wang, Q. Zhang, Z. Shen, Y. He, Z. Chen, J. Li, et al., Predicting transcription factor binding sites using DNA shape features based on shared hybrid deep learning architecture, *Mol. Ther. Nucleic Acids*, **24** (2021), 154–163. https://doi.org/10.1016/j.omtn.2021.02.014
18. Q. Zhang, Y. He, S. Wang, Z. Chen, Z. Guo, Z. Cui, et al., Base-resolution prediction of transcription factor binding signals by a deep learning framework, *PLoS Comp. Biol.*, **18** (2022), e1009941. https://doi.org/10.1371/journal.pcbi.1009941
19. S. Wang, Y. He, Z. Chen, Q. Zhang, FCNGRU: Locating transcription factor binding sites by combing fully convolutional neural network with gated recurrent unit, *IEEE J. Biomed. Health. Inf.*, **26** (2021), 1883–1890. https://doi.org/10.1109/JBHI.2021.3117616

20. Q. Zhang, Z. Shen, D. S. Huang, Predicting in-vitro transcription factor binding sites using DNA sequence+ shape, *IEEE/ACM Trans. Comput. Biol. Bioinf.*, **18** (2019), 667–676. https://doi.org/10.1109/TCBB.2019.2947461
21. Q. Zhang, S. Wang, Z. Chen, Y. He, Q. Liu, D. S. Huang, Locating transcription factor binding sites by fully convolutional neural network, *Briefings Bioinf.*, **22** (2021), bbaa435. https://doi.org/10.1093/bib/bbaa435
22. Y. Zhang, Z. Wang, Y. Zeng, Y. Liu, S. Xiong, M. Wang, et al., A novel convolution attention model for predicting transcription factor binding sites by combination of sequence and shape, *Briefings Bioinf.*, **23** (2022), bbab525. https://doi.org/10.1093/bib/bbab525
23. Y. Zhang, Z. Wang, Y. Zeng, J. Zhou, Q. Zou, High-resolution transcription factor binding sites prediction improved performance and interpretability by deep learning method, *Briefings Bioinf.*, **22** (2021), bbab273. https://doi.org/10.1093/bib/bbab273
24. Y. He, Z. Shen, Q. Zhang, S. Wang, D. S. Huang, A survey on deep learning in DNA/RNA motif mining, *Briefings Bioinf.*, **22** (2021), bbaa229. https://doi.org/10.1093/bib/bbaa229
25. W. S. Noble, S. Kuehn, R. Thurman, M. Yu, J. Stamatoyannopoulos, Predicting the in vivo signature of human gene regulatory sequences, *Bioinformatics*, **21** (2005), i338–i343. https://doi.org/10.1093/bioinformatics/bti1047
26. B. Manavalan, T. H. Shin, G. Lee, DHSpred: Support-vector-machine-based human DNase I hypersensitive sites prediction using the optimal features selected by random forest, *Oncotarget*, **9** (2018), 1944. https://doi.org/10.18632/oncotarget.23099
27. S. Zhang, W. Zhuang, Z. Xu, Prediction of DNase I hypersensitive sites in plant genome using multiple modes of pseudo components, *Anal. Biochem.*, **549** (2018), 149–156. https://doi.org/10.1016/j.ab.2018.03.025
28. Y. Liang, S. Zhang, IDHS-DMCAC: Identifying DNase I hypersensitive sites with balanced dinucleotide-based detrending moving-average cross-correlation coefficient, *SAR QSAR Environ. Res.*, **30** (2019), 429–445. https://doi.org/10.1080/1062936X.2019.1615546
29. S. Zhang, Z. Duan, W. Yang, C. Qian, Y. You, IDHS-DASTS: Identifying DNase I hypersensitive sites based on LASSO and stacking learning, *Mol. Omics*, **17** (2021), 130–141. https://doi.org/10.1039/D0MO00115E
30. B. Liu, R. Long, K. C. Chou, IDHS-EL: Identifying DNase I hypersensitive sites by fusing three different modes of pseudo nucleotide composition into an ensemble learning framework, *Bioinformatics*, **32** (2016), 2411–2418. https://doi.org/10.1093/bioinformatics/btw186
31. S. Zhang, J. Lin, L. Su, Z. Zhou, PDHS-DSET: Prediction of DNase I hypersensitive sites in plant genome using DS evidence theory, *Anal. Biochem.*, **564** (2019), 54–63. https://doi.org/10.1016/j.ab.2018.10.018
32. Y. Zheng, H. Wang, Y. Ding, F. Guo, CEPZ: A novel predictor for identification of DNase I hypersensitive sites, *IEEE/ACM Trans. Comput. Biol. Bioinf.*, **18** (2021), 2768–2774. https://doi.org/10.1109/TCBB.2021.3053661
33. S. Zhang, Q. Yu, H. He, F. Zhu, P. Wu, L. Gu, et al., IDHS-DSAMS: Identifying DNase I hypersensitive sites based on the dinucleotide property matrix and ensemble bagged tree, *Genomics*, **112** (2020), 1282–1289. https://doi.org/10.1016/j.ygeno.2019.07.017
34. S. 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. Genomics*, **295** (2020), 1431–1442. https://doi.org/10.1007/s00438-020-01711-8

35. Z. C. Xu, S. Y. Jiang, W. R. Qiu, Y. C. Liu, X. Xiao, IDHSs-PseTNC: Identifying DNase I hypersensitive sites with pseuo trinucleotide component by deep sparse auto-encoder, *Lett. Org. Chem.*, **14** (2017), 655–664. https://doi.org/10.2174/1570178614666170213102455

36. C. Lyu, L. Wang, J. Zhang, Deep learning for DNase I hypersensitive sites identification, *BMC genomics*, **19** (2018), 155–165. https://doi.org/10.1186/s12864-018-5283-8

37. P. Feng, N. Jiang, N. Liu, Prediction of DNase I hypersensitive sites by using pseudo nucleotide compositions, *Sci. World J.*, **2014** (2014), 740506. https://doi.org/10.1155/2014/740506

38. W. Chen, T. Y. Lei, D. C. Jin, H. Lin, K. C. Chou, PseKNC: A flexible web server for generating pseudo K-tuple nucleotide composition, *Anal. Biochem.*, **456** (2014), 53–60. https://doi.org/10.1016/j.ab.2014.04.001

39. W. Chen, H. Lin, K. C. Chou, Pseudo nucleotide composition or PseKNC: An effective formulation for analyzing genomic sequences, *Mol. Biosyst.*, **11** (2015), 2620–2634. https://doi.org/10.1039/C5MB00155B

40. B. Liu, F. Liu, X. Wang, J. Chen, L. Fang, K. C. Chou, Pse-in-One: A web server for generating various modes of pseudo components of DNA, RNA, and protein sequences, *Nucleic Acids Res.*, **43** (2015), W65–W71. https://doi.org/10.1093/nar/gkv458

41. S. Zhang, Z. Zhou, X. Chen, Y. Hu, L. Yang, PDHS-SVM: A prediction method for plant DNase I hypersensitive sites based on support vector machine, *J. Theor. Biol.*, **426** (2017), 126–133. https://doi.org/10.1016/j.jtbi.2017.05.030

42. K. He, X. Zhang, S. Ren, J. Sun, Spatial pyramid pooling in deep convolutional networks for visual recognition, *IEEE Trans. Pattern Anal. Mach. Intell.*, **37** (2015), 1904–1916. https://doi.org/10.1109/TPAMI.2015.2389824

43. F. Y. Dao, H. Lv, W. Su, Z. J. Sun, Q. L. Huang, H. Lin, IDHS-deep: an integrated tool for predicting DNase I hypersensitive sites by deep neural network, *Briefings Bioinf.*, **22** (2021), bbab047. https://doi.org/10.1093/bib/bbab047

44. C. E. Breeze, J. Lazar, T. Mercer, J. Halow, I. Washington, K. Lee, et al., Atlas and developmental dynamics of mouse DNase I hypersensitive sites, *bioRxiv*, **2020** (2020). https://doi.org/10.1101/2020.06.26.172718

45. W. Li, A. Godzik, Cd-hit: A fast program for clustering and comparing large sets of protein or nucleotide sequences, *Bioinformatics*, **22** (2006), 1658–1659. https://doi.org/10.1093/bioinformatics/btl158

46. L. Fu, B. Niu, Z. Zhu, S. Wu, W. Li, CD-HIT: Accelerated for clustering the next-generation sequencing data, *Bioinformatics*, **28** (2012), 3150–3152. https://doi.org/10.1093/bioinformatics/bts565

47. X. Tang, P. Zheng, X. Li, H. Wu, D. Q. Wei, Y. Liu, et al., Deep6mAPred: A CNN and Bi-LSTM-based deep learning method for predicting DNA N6-methyladenosine sites across plant species, *Methods*, **204** (2022), 142–150. https://doi.org/10.1016/j.ymeth.2022.04.011

48. T. Mikolov, K. Chen, G. Corrado, J. Dean, Efficient estimation of word representations in vector space, preprint, arXiv:1301.3781.

49. T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, J. Dean, Distributed representations of words and phrases and their compositionality, in *Advances in neural information processing systems*, **26** (2013), 3111–3119.

50. K. Fukushima, S. Miyake, Neocognitron: A new algorithm for pattern recognition tolerant of deformations and shifts in position, *Pattern Recognt.*, **15** (1982), 455–469. https://doi.org/10.1016/0031-3203(82)90024-3
51. D. H. Hubel, T. N. Wiesel, Receptive fields, binocular interaction and functional architecture in the cat's visual cortex, *J. Physiol.*, **160** (1962), 106. https://doi.org/10.1113/jphysiol.1962.sp006837
52. Y. LeCun, B. Boser, J. Denker, D. Henderson, R. Howard, W. Hubbard, et al., Handwritten digit recognition with a back-propagation network, in *Advances in neural information processing systems*, Morgan Kaufmann, **2** (1989), 396–404.
53. S. Hochreiter, J. Schmidhuber, Long short-term memory, *Neural Comput.*, **9** (1997), 1735–1780. https://doi.org/10.1162/neco.1997.9.8.1735
54. A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, et al., Attention is all you need, in *Advances in neural information processing systems*, **30** (2017), 6000–6010.
55. C. Raffel, D. P. Ellis, Feed-forward networks with attention can solve some long-term memory problems, preprint, arXiv:1512.08756.