Computational and Structural Biotechnology Journal 19 (2021) 1612–1619

COMPUTATIONAL AND STRUCTURAL BIOTECHNOLOGY JOURNAL

journal homepage: www.elsevier.com/locate/csbj

Identifying DNA N4-methylcytosine sites in the rosaceae genome with a deep learning model relying on distributed feature representation ☆

Jhabindra Khanal [a], Hilal Tayara [b], Quan Zou [c,*], Kil To Chong [a,d,*]

[a] Department of Electronics and Information Engineering, Jeonbuk National University, Jeonju 54896, South Korea
[b] School of international Engineering and Science, Jeonbuk National University, Jeonju 54896, South Korea
[c] Institute of Fundamental and Frontier Sciences, University of Electronic Science and Technology of China, Chengdu 610054, China
[d] Advanced Electronics and Information Research Center, Jeonbuk National University, Jeonju 54896, South Korea

ARTICLE INFO

Article history:
Received 9 January 2021
Received in revised form 12 March 2021
Accepted 13 March 2021
Available online 19 March 2021

Keywords:
Sequence analysis
DNA N4-methylcytosine (4mC)
Word embedding
Convolutional Neural Network
Web-server

ABSTRACT

DNA N4-methylcytosine (4mC), an epigenetic modification found in prokaryotic and eukaryotic species, is involved in numerous biological functions, including host defense, transcription regulation, gene expression, and DNA replication. To identify 4mC sites, previous computational studies mostly focused on finding hand-crafted features. This area of research, therefore, would benefit from the development of a computational approach that relies on automatic feature selection to identify relevant sites. We here report 4mC-w2vec, a computational method that learned automatic feature discrimination in the *Rosaceae* genomes, especially in *Rosa chinensis (R. chinensis)* and *Fragaria vesca (F. vesca)*, based on distributed feature representation and through the word embedding technique 'word2vec'. While a few bioinformatics tools are currently employed to identify 4mC sites in these *genomes*, their prediction performance is inadequate. Our system processed 4mC and non-4mC sites through a word embedding process, including sub-word information of its biological words through k-mer, which then served as features that were fed into a double layer of convolutional neural network (CNN) to classify whether the sample sequences contained 4mCs or non-4mCs sites. Our tool demonstrated performance superior to current tools that use the same genomic datasets. Additionally, 4mC-w2vec is effective for balanced and imbalanced class datasets alike, and the online web-server is currently available at: http://nsclbio.jbnu.ac.kr/tools/4mC-w2vec/.

1. Introduction

Epigenetics refers to the heritable changes in gene function that are not related to modifications of the DNA sequence itself [1]. DNA methylation is one of the most widely known epigenetic marks, as it plays a vital role in various critical biological process, including changes in chromatin structure, ensuring the stability of DNA, gene-expression control, DNA conformation, X-chromosome inactivation, gene regulation, cellular differentiation, and cancer progression [2–5]. One of the most widespread DNA methylation modification is N4-methylcytosine (4mC), it was primarily described in 1983 [6] which is methylated on the fourth position of the cytosine pyrimidine ring of both eukaryotes and prokaryotes (though 4mC is more commonly found and studied in the latter). In prokaryotes, 4mC is part of a restriction-modification(R-M) system that defends against activities of foreign DNA, including its repair, expression, and replication [7–11]. 4mC also plays a supplementary role in, among other things, genome stabilization, recombination, and evolution [12–14]. The biological roles of 4mC in eukaryotes is less understood, in part because the small size of 4mC in the eukaryote genome prevents its detection through anything other than high sensitivity techniques.

To identify 4mC sites experimentally, Single Molecule of Real-Time (SMRT), mass spectrometry, and methylation-precise PCR have all been used [15–18]. These methods, however, are time consuming and labor-intensive. Analysis of the 'big data' associated with the *Rosaceae genome*, with proper computational tools may be a more efficient means of accurately identifying 4mC sites. Several in silico methods have been proposed to identify 4mC sites for

☆ Jhabindra Khanal and Hilal Tayara contributed equally.

* Corresponding authors at: Department of Electronics and Information Engineering, Jeonbuk National University, Jeonju 54896, South Korea (K.T. Chong); Institute of Fundamental and Frontier Sciences, University of Electronic Science and Technology of China, Chengdu 610054, China (Quan Zou).

E-mail addresses: zouquan@nclab.net (Q. Zou), kitchong@jbnu.ac.kr (K.T. Chong).

https://doi.org/10.1016/j.csbj.2021.03.015

some species (e.g. E. coli, G. subterraneus, A. thaliana, D. melanogaster, C. elegans, G. pickeringii, and mice) using the recently constructed database MethSMRT [19–32]. To the best of our knowledge, only two computational methods are currently available to identify 4mC sites in the Rosaceae genome: i4mC-Rose [33] and DNC4mC-Deep [34]. The i4mC-Rose tool is the result of a random forest classifier with multiple encoding schemes, while DNC4mC-Deep is the result of a deep learning approach with six encoding techniques. Although these methods produced acceptable results, there is still much room for improvement, especially given that the adopted datasets may not have been of sufficient quality to capture the 4mC motifs, or the feature selection methods employed may not have been suitable to distinguish between the sequence information of positive and negative classes. Moreover, previous methods relied on domain knowledge to hand-design for the input features. Our method, in contrast, captures automatically a high level of input features by word embedding, allowing for a novel and highly accurate computational tool.

In this paper, a sequence-based DNA 4mC sites predictor was developed. Our central idea was to transform the DNA sequences into vectors by word embedding and then process these with a double-layer one-dimensional CNN for the final classification. Word embedding was invented to apply in by Google in 2013 [35] to assist with natural language processing (NLP), but it later found success in number of biological applications [36–43], deep learning of the sort employed in our second step has achieved notable results in a number of areas, including speech recognition [44], image recognition [45,46], NLP [47], and genome wide prediction [48–53]. In our study, integrating the techniques of word embedding and deep learning gave outstanding results for both balanced and imbalanced class datasets, and we suggest that the proposed method is promising for genome-wide prediction.

2. Materials and methods

2.1. Datasets construction

It was necessary to construct a reliable dataset to develop our sequence-based identifier. We independently constructed a complete set of training and independent datasets. 4mC containing sequences (the positive sequences) were obtained from the MDR database [54], http://mdr.xieslab.org/. According to previous researches, the best prediction performances were obtained with the length of 41-nt [22,29]. Therefore, the length of the DNA sequences were set to 41-nt, containing 'C' at the center. Previous researchers [33,34] applied a modification QV (modQV) score of $\geqslant$20 to generate a positive dataset, but as W.Chen et al. have pointed out, a modQV score of 30 or more is the default or the best threshold for labelling the position of a cytosine as modified [21]. In the interest of developing a more reliable model, we applied QV of $\geqslant$30 to construct our positive dataset and excluded the sequences that share QV values $<$30. To remove sequence similarity, CD-HIT [55] software with the cutoff threshold of 65.00% was used. As a result of these procedures, we obtained 4321 in *F. vesca* genome, and 2421 positive sequences in *R. chinensis* genome. From these datasets, approximately 80% of the sequences (3457 (*F. vesca*) and 1938 (*R. chinensis*)) were selected as training sets, with remaining sequences (864 (*F. vesca*) and 483 (*R. chinensis*)), used as independent datasets.

The negative sequences (non-4mC site containing sequences) were obtained from the same genome file where the 4mC sites ('C' at the center) was not detected by the SMRT sequencing technique. In this way, a large number of negative sequences in each species were formed with 'C' at the center. For model training, positive and negative sequences were balanced out. To test the efficiency of our proposed model, we constructed the independent datasets with different ratios of positive and negative samples. For *F. vesca* these were: 1:1 [864 positive and 864 negative sequences], 1:5 [864 positive and 4320 negative sequences], and 1:15 [864 positive and 12960 negative sequences]. For R. chinensis, 1:1 [483 positive and 483 negative sequences], 1:5 [483 positive and 2415 negative sequences], and 1:15 [483 positive and 7245 negative sequences]. Due to limit number of the independent-positive sequences the same positive sequences were accepted for all ratio groups (i.e. 864 for *F. vesca* and 483 for *R. chinensis*). The negative sequences did not overlap across ratio groups. These training and independent datasets for both species is summarized in Table 1.

We elected to construct such imbalanced class datasets prior to testing as it is common to find real-world datasets that have such strongly imbalanced distributions. Accordingly, we aimed that to assist researchers with testing imbalanced datasets using a classifier. To the best of our knowledge, we first to proposed an i4mC-w2vec tool that deal with imbalanced class datasets in this area (4mC sites prediction).

2.2. Methodology

We present a novel method (4mC-w2vec) for identifying 4mC sites in the Rosaceae genome. Our consists of two major steps. The first step is the discriminative feature generation or representation stage in which each DNA sequence is described into words using 3-mer, after which a word-embedding method is applied to map each word to its corresponding feature representation. For the second step, a deep learning model is used to classify 4mCs and non-4mCs based on the generated features of the first stage. A detailed explanation is presented in the following sections, and the general architecture is illustrated in Fig. 1.

2.2.1. Distributed feature representation

These days many real-world biological data applications involve datasets that are strongly imbalanced distributive, complex, and noisy. We decided to apply a word embedding technique commonly known as 'word2vec' [35]. This technique generates an optimal set of feature vectors based on distributional hypothesis [56]. Word2vec is a two-layer neural network that processes text by vectorizing words as depicted in Fig. 1 (a). It receives input as a text corpus and its output is feature vectors that represent words in that corpus. This technique decreases computational complexity and reduce the noise, ultimately leading to improved performance in the resultant computational model. Additionally, many biological codes (such as genetic code) can be represented as a language [57–59], with the resulting insights can being applied towards solving a variety of biological problems [58,60,61]. Accordingly, we adopted the word2vec method to find interpretable representations for each 4mC sites.

Corpus construction discovers the semantic relations between words large files. For our research, we generated the corpus by processing the genomes of *F. vesca* (wild strawberry (NC_020491.1)) and *R. chinensis* (Chinese rose (NC_037093.1)) using NCBI genomic data, available at https://www.ncbi.nlm.nih.gov. The first step of training word2vec is building a corpus vocabulary. The word2vec

Table 1
Summary of training and independent test datasets for *F. vesca* and *R. chinensis*.

Genomes	Positive/Negative	Training datasets	Independent datasets
F. vesca	Positive	3457	864
	Negative	3457	864, 4320, 12960
R. chinensis	Positive	1938	483
	Negative	1938	483, 2415, 7245

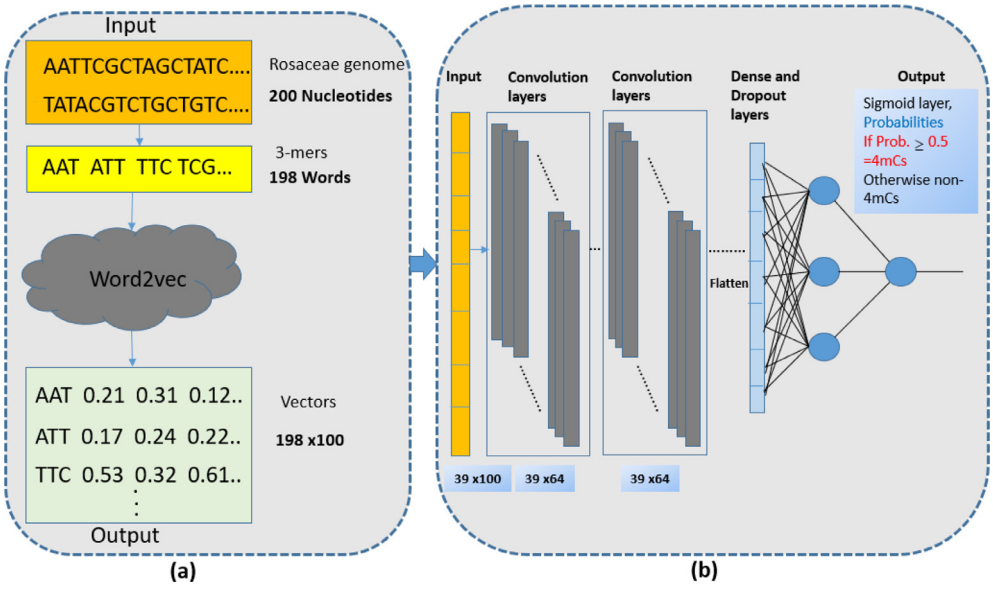

Fig. 1. A general architecture of the proposed model: (a) word embedding process and (b) one-dimensional CNN model.

model can be applied based on either Continuous Bag-Of-Words (CBOW) or Skip-gram methods. In the Skip-gram model, the current word (w(t)) or input is used to predict the surrounding window of context word. In contrast, the CBOW method attempts to guess the target word based on its neighboring (context) words. As inputs into a CBOW model, a window size of five was formulated as follows Eqn 1:

$$\sum_{k=-2,k\neq 0}^{2} w(t+k) \tag{1}$$

The CBOW and Skip-gram perform similarly, although Skip-gram is more useful and gives a better outcome for infrequent words [62]. In our study, we are concerned with frequent words, and therefore adopted CBOW for word2vec training. To process the CBOW, genome assemblies were divided into sentences with lengths of 200-nt. Next, each sentence was divided into overlapping 3-mer to form words (such as AAT, CCT, GCN, and CCC). At this point, each 4mC contains a chain of continuous nucleotides. Those words were fed into a two layer of word2vec model as depicted in Fig. 1 (a). As a result, each word had its own 100-dimension (D) vector representation, with each sequence of length *L*represented by an array of shape $(L-2)\times 100$. For example, the word 'AAT' was represented as a 100-(D) vector of $[0.11_1, 0.22_2, 0.33_3, \ldots\ldots\ldots, 0.12_{100}]$and 'CCT' was represented as a 100-D vector of $[0.22_1, 0.11_2, 0.31_3, \ldots\ldots, 0.23_{100}]$. The parameters that were used to train word2vec are listed in Table 2. Most of the parameters were left as default. According to the previous researches the best performance was obtained by the creation of 100-D [36,43,50]. Therefore, the parameter for dimension of the word vectors was set to 100-D. We included all words with frequency greater than 1. For context words we have tested different overlapping k-mers such as k = 1 (A), k = 2 (AT), k = 4 (ATCG), k = 5 (ATCGA), and k = 6 (ATCGAT). Negative sampling was set to 5 to draw 'noise words'. Window size was set to 5 for maximum distance between the current and predicted word within a sentence. Number of epochs (iterations) over the corpus was set to 20. The word2vec was trained independently for both species using the python library genism [63].

2.2.2. CNN model

As shown in Fig. 1 (b), we used a CNN model (a deep feed-forward neural network) to learn the features generated from word2vec. In a CNN, hyper-parameters determine layer architecture in the training step, and this affects model accuracy and learning time. Therefore, grid search strategy was used for hyper-parameters optimization, including the number of filters, the kernel sizes (size of filters), the dropout rates, the number of convolutional layers, and the activation functions. After applying the grid search technique, the proposed CNN model yielded two one dimensional convolution layers with 64 filters of 9 units and one stride unit. In each convolution layer, a rectified linear activation unit (ReLU) was used as an activation function. To fix the overfitting problem, the first layer convolution was followed by a dropout layer with a rate of 0.7. For final classification, a fully connected layer with one node followed by sigmoid function was used. The configuration of the CNN model is presented in Table 3. To train the CNN model on the training datasets, the learning rate was set

Table 2
Word2vec training parameters.

Parameters	word2vec model
Training Method	CBOW
Vector Size	100
Corpus	Genomes of F. vesca and R. chinensis
Minimum Count	1
Context Words	3-mer
Negative Sampling	5
Window Size	5
Number of Epochs	20

Table 3
The proposed CNN's architecture.

Layers	Output shape
Input	[39, 100]
Conv1D (64,9,1)	[39, 64]
Conv1D (64,9,1)	[39, 64]
Dropout (0.5)	[1248]
Dense	[1]
Sigmoid	[1]

to 0.0007, and the batch size was set to 128 with an early stopping strategy based on the validation loss. RMSprop was used as an optimizer [64] and binary cross-entropy was used as a loss function [65]. The Keras framework, a python open source library, (https://keras.io/), was used to build the 4mC-w2vec. The trained models will be able to learn an imbalanced class datatsets by setting the 'class weight' during the CNN training phase. Therefore, we used the 'class weight' programmatically using the Scikit-learn [66].

2.3. Evaluation parameters

Various statistical metrics, including sensitivity (Sn), specificity (Sp), accuracy (ACC), and Matthew correlation coefficient (MCC) were used to evaluate the performance of the models [67–69].

$$Sn = 1 - \frac{M_-^+}{M^+},\ 0 \leqslant Sn \leqslant 1 \tag{2}$$

$$Sp = 1 - \frac{M_+^-}{M^-},\ 0 \leqslant Sp \leqslant 1 \tag{3}$$

$$ACC = 1 - \frac{M_-^+ + M_+^-}{M^+ + M^-},\ 0 \leqslant ACC \leqslant 1 \tag{4}$$

$$MCC = \frac{1 - \frac{M_-^+ + M_+^-}{M^+ + M^-}}{\sqrt{(1 + \frac{M_+^- - M^+}{M^+})(1 + \frac{M_-^+ - M_+^-}{M^-})}},\ -1 \leqslant MCC \leqslant 1 \tag{5}$$

The symbols in Eqs. (2)–(5) are:

$$\begin{cases} M_+^- & = FP \\ M_-^+ & = FN \\ M^+ & = TP + M_-^+ \\ M^- & = TN + M_+^- \end{cases}$$

where, TP, FP, TN, FN are either true positive, false positive, true negative, and false negative values. We also included the Receiver operating Characteristic (ROC) curves to evaluate the proposed method. Overall performance quality was represented by the area under the ROC curve (auROC) [70]. When evaluating binary classifiers on imbalanced class datasets, the precision-recall curve is more helpful than the ROC curve (as pointed out by [71]).

3. Result and discussion

3.1. Analysis of nucleotide composition preference

To demonstrate the nucleotide composition preferences between positives (4mC containing sequences) and negatives (non-4mC containing sequences), the Two Sample Logo tool was used [72]. The height of bases was formed as maintained by their statistical significance ($p \leq 0.05$by t-test). As seen in Fig. 2, the 'C' nucleobase was located in the center of the sequences with length 41. In case of *F. vesca*, both the C and G bases were enriched (over-represented), while both 'A' and 'T' bases were depleted (under-representated). Specifically, 'C' was over-represented at positions 1–3, 7–12, 14–20, 23, 24, 27, 30, and 33–41 and under-represented at position at 6, while 'G' base was enriched at positions 1, 4–7, 10, 11, 13–20, 22–26, 29, 32, 34, 35, 37, and 41 and depleted at position 28. The 'A' base was depleted at positions 1–3, 7–19, 22–24, 27, 30, and 34–39 and enriched at positions 25, 26, 28, and 29, while 'T' was depleted at positions 1, 3, 4, 7–11, 13–20, 22–27, 29, 32, 33, 35, 37, 38, 40, and 41, and was not enriched at any position. Some nucleotide base pairs became visible along the DNA sequences. For example, in the 4mC containing sequences two consecutive 'C' and 'G' bases were spotted at positions 1–3, 7, 11, 14, 14–20, 23, 24 34, 35, and 37–41.

In case of *R. chinensis*, 'A' was enriched at positions 4, 25, 26, 28, 29, and 33 and depleted at positions 20, 22, and 23. The successive 'C' was enriched at positions 7, 12–20, 23, 27, 30, 38, and 39 and depleted at positions 5, 6, 25, and 26. 'G' was enriched at positions 5, 6, 20, 22–24, 26, and 35 and depleted at position 4, 12, 15, 16, 19, and 28–30, while 'T' was enriched in only a few positions, includ-

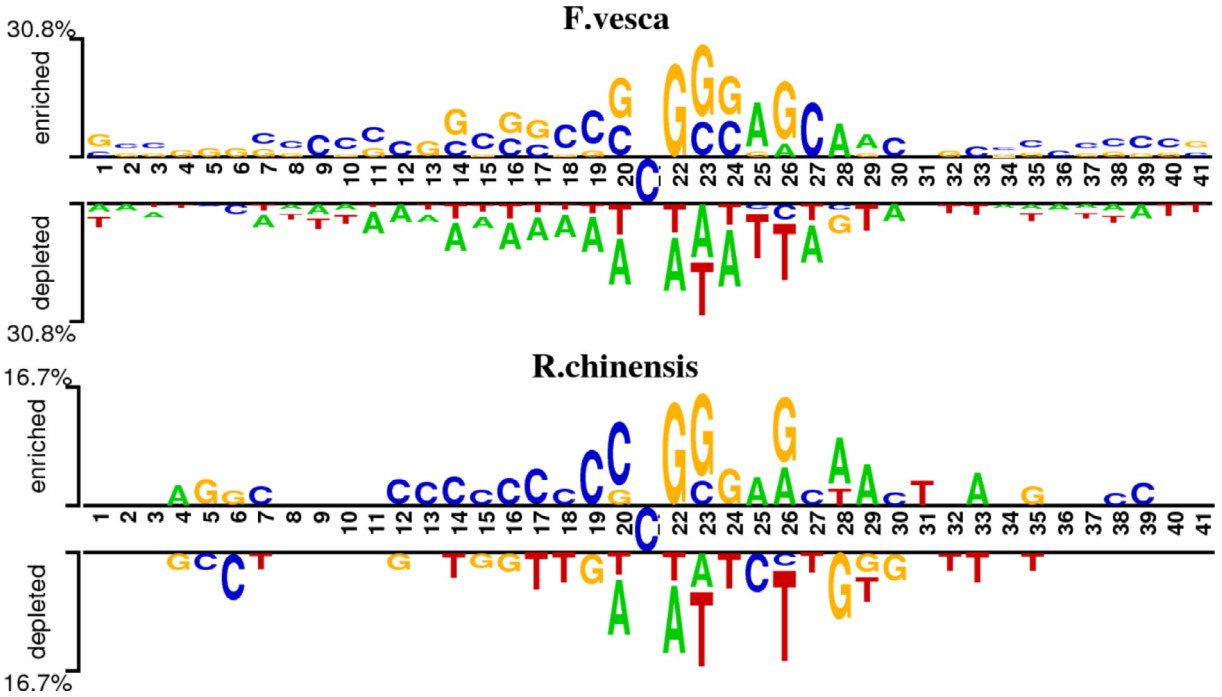

Fig. 2. Demonstration of nucleotide composition preferences between positives (4mC containing sequences) and negatives (non-4mC containing sequences) for *F. vesca* and *R. chinensis* datasets.

ing 28, and 31 and depleted at positions 7, 14, 17, 18, 20, 22–24, 26, 27, 29, 32, 33, and 35. To put it succinctly, in both species, there was significant variation between over-represented and under-represented nucleotides between the 4mC and non-4mC containing sequences.

All results shown in Fig. 2 demonstrate that the four nucleotides distribution around 4mC sites has statistically significant position-specific difference between 4mC containing and non-4mC containing samples. Therefore, it is possible to design a computational model to identify 4mC sites only based on sequence information.

3.2. Effect of using different encoding methods

Based on overlapping k-mer values (such as 1-mer, 2-mer, 3-mer, 4-mer, 5-mer, and 6-mer) six-feature vectors models were obtained by the word embedding process. All these vector representation model were fed into the CNN for independent identification of 4mC sites. We observed that the 3-mer was more informative for predicting 4mC sites for both species. In this study, the word2vec representation based on 3-mer and classified by CNN was considered the final model, or the 'i4mC-w2vec'. In cross-validation test, the proposed predictor obtained 0.7407 MCC, 0.8697 accuracy, and 0.9400 AUC for the *F. vesca*, and 0.7093 MCC, 0.8541 accuracy, and 0.9370 AUC the predictor obtained for *R. chinensis*.

According to prior research, biological sequences encoded with one-hot method in conjunction with deep learning model performed well in 4mC prediction task [20]. We accordingly used one-hot encoding scheme to encode the DNA sequences, in which nucleotides A, C, G, and T were coded as (1 0 0 0), (0 1 0 0), (0 0 1 0), and (0 0 0 1), respectively. To determine the best parameters for CNN using one-hot encoding, the grid search algorithm was used. The results showed that the word2vec (based on 3-mer and 4-mer) method outperformed the one-hot method. The performance of the six word embedding model based on different k-mers and one-hot encoding when classified by CNN is presented in Table 4. More generally, the auROC of the *F. vesca* was 0.8920 using one-hot encoding but 0.9400 using word2vec (3-mer) encoding (Fig. 3) (a). Similarly, auROC of the *R. chinensis* was 0.9110 using one-hot encoding while it is 0.9370 using word2vec (3-mer) (Fig. 3) (b).

3.3. Performance comparison with existing methods on the independent test datasets

To test whether the 4mC-w2vec could identify 4mC sites on balanced and imbalanced blind datasets, we ran the model on the independent test datasets with different ratios of positive and negative samples (see Section 2.1). For imbalanced classification with a few sequences of minority (positive) class, auROC can be misleading, a large change in a ROC curve or auROC score

Table 4
Performance of the CNN using different word2vec models (based on k-mers) and one-hot encoding on the training dataset for both species by a 5-fold cross-validation test.

Species	Methods	Sn	Sp	ACC	MCC	AUC
F. vesca	k = 1	0.7963	0.7700	0.7832	0.5666	0.8520
	k = 2	0.7984	0.8295	0.8141	0.6283	0.8141
	k = 3	**0.8976**	0.8417	**0.8697**	**0.7407**	**0.9400**
	k = 4	0.8141	**0.8600**	0.8374	0.6751	0.9155
	k = 5	0.7931	0.7582	0.7754	0.5516	0.8505
	k = 6	0.7984	0.7302	0.7638	0.5296	0.8435
	onehot	0.8507	0.8244	0.8374	0.6752	0.8920
R. chinensis	k = 1	**0.8711**	0.6873	0.7793	0.5682	0.8781
	k = 2	0.8144	0.8199	0.8541	0.6335	0.8934
	k = 3	0.8219	**0.8854**	**0.8541**	**0.7093**	**0.9370**
	k = 4	0.8664	0.7633	0.8141	0.6326	0.8891
	k = 5	0.8220	0.7519	0.7870	0.5755	0.8755
	k = 6	0.7722	0.8066	0.7896	0.5793	0.8604
	onehot	0.7958	0.8371	0.8167	0.6337	0.9110

Note: The best performance value for each metric across different methods is highlighted in bold.

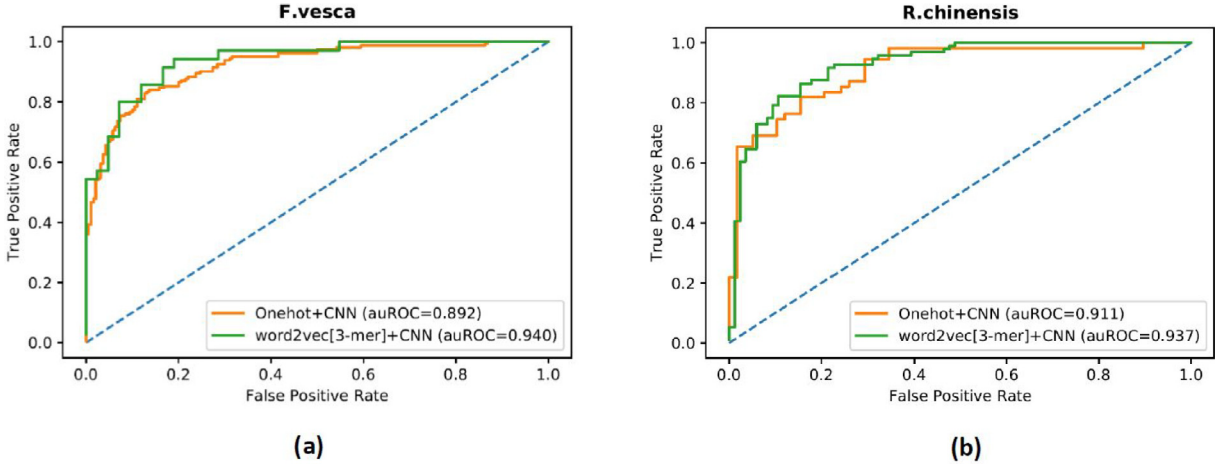

Fig. 3. Performance comparisons of word2vec-based model and one-hot encoding-based model when classified by CNN using a 5-fold cross-validation test on *F. vesca* (a) and *R. chinensis* (b).

may occur with even a small number of correct or incorrect predictions made by a model [71,73]. For this reason, numerous surveys have suggested that a precision-recall curve (PR curve) is a superior alternative [74]. A PR curve is a plot of the precision (y-axis) and recall (x-axis) of different probability thresholds. Precision and recall are concerned on minority class (positive), but not majority (negative) class [75]. A precision-recall AUC (PRauc) score of 1 represents a perfect model.

To demonstrate the superiority of i4mC-w2vec method, a comparison with existing methods was performed, including i4mC-Fuse [30], and DNC4mC-Deep [31]. These two web-servers were recently constructed, and both focus on the genomes of *F. vesca* and *R. chinensis* to identify 4mC cites. For our comparison, we directly submitted the same positive/negative ratios of the independent datasets to these two web-servers. The performance of the 4mC-w2vec, DNC4mC-Deep, and i4mC-Fuse based on different evaluation indexes is presented in Table 5, with corresponding precision-recall curves presented in Fig. 4. As shown in Table 5, between these three methods, 4mC-w2vec achieved the best performance, as measured across all evaluation indexes for all the different ratios of the both species.

It is clear that our model performed better than the existing ones for every ration group. Specifically, PR AUCs of the 4mC-w2vec were 3–6% higher than those of the two existing methods,

Table 5
The performance of the i4mC-Fuse, DNC4mC-Deep, and i4mC-w2vec on the independent datasets with different ratios.

Species	Method	Sn	Sp	ACC	MCC	PRauc
F. vesca	**i4mC-Fuse**					
	ratio of [1:1]	0.8376	0.7209	0.7793	0.5624	0.8482
	ratio of [1:5]	0.8530	0.7105	0.7819	0.5695	0.8606
	ratio of [1:15]	0.8569	0.6434	0.7703	0.5586	0.8517
	DNC4mC-Deep					
	ratio of [1:1]	0.8582	0.7390	0.7987	0.6016	0.8438
	ratio of [1:5]	0.8560	0.6950	0.7858	0.5810	0.8694
	ratio of [1:15]	0.8556	0.7183	0.7870	0.5795	0.8723
	i4mC-w2vec					
	ratio of [1:1]	0.8994	0.8268	0.8632	0.7283	0.9176
	ratio of [1:5]	0.8814	0.8449	0.8632	0.7269	0.9188
	ratio of [1:15]	0.8762	0.8062	0.8412	0.6842	0.9021
R. chinensis	**i4mC-Fuse**					
	ratio of [1:1]	0.8505	0.7312	0.7909	0.5860	0.8646
	ratio of [1:5]	0.8411	0.6718	0.7716	0.5541	0.8526
	ratio of [1:15]	0.8072	0.6149	0.7612	0.5461	0.8507
	DNC4mC-Deep					
	ratio of [1:1]	0.8637	0.7131	0.7935	0.5946	0.8700
	ratio of [1:5]	0.8537	0.7235	0.7987	0.6000	0.8641
	ratio of [1:15]	0.8391	0.6511	0.7703	0.5564	0.8594
	i4mC-w2vec					
	ratio of [1:1]	0.8737	0.8242	0.8490	0.6988	0.9099
	ratio of [1:5]	0.884	0.7957	0.8400	0.6825	0.8966
	ratio of [1:15]	0.8940	0.8113	0.8477	0.6972	0.9136

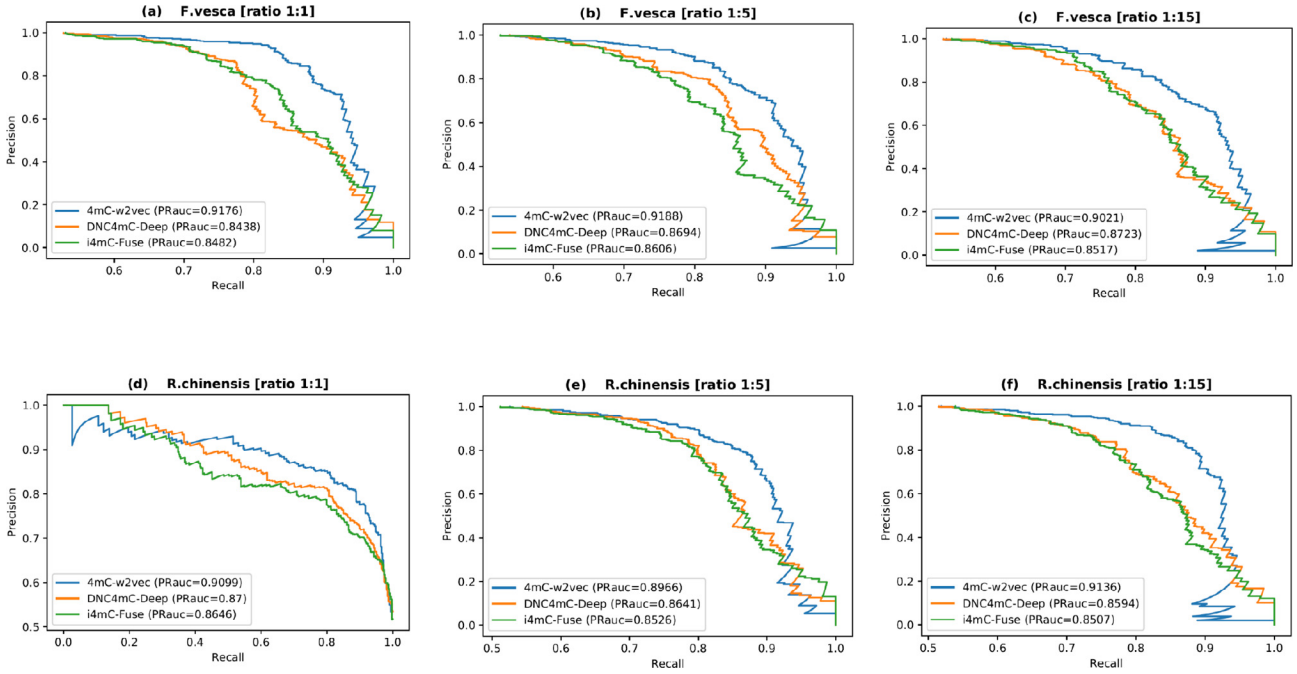

Fig. 4. comparison of PRC generated by our method and two existing methods on the different ratios of the balanced/imbalanced independent test datasets for both species. The PRauc scores and PR curves show that the 4mC-w2vec outperforms the existing methods in the *F. vesca* (a–c) and *R. chinensis* (d–e) datasets.

showing that our model is the most appropriate for 4mC site prediction on both imbalanced (ratio groups 1:5,1:15) and balanced (ratio group 1:1) datasets. Moreover, our results demonstrate that our model is stable against the increasing ratios of the imbalanced class datasets, while the performance of other methods decreased as the positive-to-negative ratio within the datasets increased.

The leading reasons for superior performance obtained from our model are as follows. Previous methods required encoding the features manually based on the domain-knowledge experience. On the other hand, the proposed model does not require any domain-knowledge. Instead, it learns the features automatically using word2vec model from the complete genome instead of using the small set of sequences. Furthermore, The input sequence of the CNN model should be encoded in a way that preserves its information. Therefore, encoding each input sequence based on the information learned from the whole genome using word2vec helped in better representation of the input sequence compared to other simple techniques such as one hot encoding as shown in Table 4.

4. Web-server

A freely accessible web application was established at http://nsclbio.jbnu.ac.kr/tools/4mC-w2vec/. The general steps to use this are: (1) upload or copy/paste the exact 41nt DNA sequence in FASTA format (sequences start with >symbol); (2) select a threshold value between 0–1 [0.5 is recommended]; (3) select a species from the list box; (4) click the 'Submit sequences' button to obtain a prediction.

The complete datasets used in this study and trained word2vec models (total 12 models, six for each species using k = 1–6) of the genomes of *F. vesca* and *R. chinensis* are available in the dataset section of the webserver http://nsclbio.jbnu.ac.kr/tools/4mC-w2vec/.

5. Conclusion

Accurately identifying 4mC sites is an important step towards understanding many biological functions. We developed a computational model using word embedding method in conjunction with a deep neural network to identify such sites. The chief advantage of the proposed model over its predecessors is the automatic creation of high dimension word-vectors for the whole genomes of *F. vesca* and *R. chinensis*, resulting in superior feature representation of 4mC sites. Put differently, the CNN can effectively capture feature generated by the word embedding process. Ultimately, our proposed method achieved better outcomes in identifying 4mC sites in both balanced and imbalanced class labels than the state-of-the-art predictors.The study presented in the paper could helpful for more widespread bioinformatics applications.

Funding

This work was supported in part by the National Research Foundation of Korea(NRF) grant funded by the Korea government(MSIT) (No. 2020R1A2C2005612) and in part by the Brain Research Program of the National Research Foundation (NRF) funded by the Korean government (MSIT) (No. NRF-2017M3C7A1044816) and in part by research funds for newly appointed professors of Jeonbuk National University, South Korea, in 2020.

CRediT authorship contribution statement

Jhabindra Khanal: Conceptualization, Data curation, Formal analysis, Methodology, Software, Validation, Visualization, Investigation, Writing - original draft, Writing - review & editing. **Hilal Tayara:** Conceptualization, Data curation, Formal analysis, Methodology, Software, Validation, Visualization, Investigation, Writing - original draft, Writing - review & editing. **Quan Zou:** Writing - original draft, Writing - review & editing. **Kil To Chong:** Writing - original draft, Writing - review & editing, Project administration, Supervision, Resources, Funding acquisition.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgement

aaa

References

[1] Moore LD, Le T, Fan G. Dna methylation and its basic function. Neuropsychopharmacology 2013;38(1):23–38.
[2] Robertson KD. Dna methylation and human disease. Nat Rev Genet 2005;6(8):597–610.
[3] Suzuki MM, Bird A. Dna methylation landscapes: provocative insights from epigenomics. Nat Rev Genet 2008;9(6):465–76.
[4] Laird PW. Principles and challenges of genome-wide dna methylation analysis. Nat Rev Genet 2010;11(3):191–203.
[5] Jones PA. Functions of dna methylation: islands, start sites, gene bodies and beyond. Nat Rev Genet 2012;13(7):484–92.
[6] Janulaitis A, Klimašauskas S, Petrušyte M, Butkus V. Cytosine modification in dna by bcni methylase yields n 4-methylcytosine. FEBS Lett 1983;161(1):131–4.
[7] Schweizer HP. Bacterial genetics: past achievements, present state of the field, and future challenges. Biotechniques 2008;44(5):633–41.
[8] Ehrlich M, Wilson G, Kuo K, Gehrke C. N4-methylcytosine as a minor base in bacterial dna. J Bacteriol 1987;169(3):939–43.
[9] Glickman BW, Radman M. Escherichia coli mutator mutants deficient in methylation-instructed dna mismatch correction. Proc Natl Acad Sci 1980;77(2):1063–7.
[10] Lu A-L, Clark S, Modrich P. Methyl-directed repair of dna base-pair mismatches in vitro. Proc Natl Acad Sci 1983;80(15):4639–43.
[11] Pukkila PJ, Peterson J, Herman G, Modrich P, Meselson M. Effects of high levels of dna adenine methylation on methyl-directed mismatch repair in escherichia coli. Genetics 1983;104(4):571–82.
[12] Vasu K, Nagaraja V. Diverse functions of restriction-modification systems in addition to cellular defense. Microbiol Mol Biol Rev 2013;77(1):53–72.
[13] Modrich P. Mechanisms and biological effects of mismatch repair. Annu Rev Genet 1991;25(1):229–53.
[14] Cheng X. Dna modification by methyltransferases. Curr Opin Struct Biol 1995;5(1):4–10.
[15] Flusberg BA, Webster DR, Lee JH, Travers KJ, Olivares EC, Clark TA, Korlach J, Turner SW. Direct detection of dna methylation during single-molecule, real-time sequencing. Nat Methods 2010;7(6):461.
[16] Doherty R, Couldrey C. Exploring genome wide bisulfite sequencing for dna methylation analysis in livestock: a technical assessment. Front Genet 2014;5:126.
[17] Boch J, Bonas U. Xanthomonas avrbs3 family-type iii effectors: discovery and function. Annu Rev Phytopathol 48..
[18] Buryanov YI, Shevchuk T. Dna methyltransferases and structural-functional specificity of eukaryotic dna modification. Biochemistry (Moscow) 2005;70(7):730–42.
[19] Ye P, Luan Y, Chen K, Liu Y, Xiao C, Xie Z. Methsmrt: an integrative database for dna n6-methyladenine and n4-methylcytosine generated by single-molecular real-time sequencing. Nucl Acids Res (2016) gkw950..
[20] Khanal J, Nazari I, Tayara H, Chong KT. 4mccnn: Identification of n4-methylcytosine sites in prokaryotes using convolutional neural network. IEEE Access 2019;7:145455–61.
[21] Chen W, Yang H, Feng P, Ding H, Lin H. idna4mc: identifying dna n4-methylcytosine sites based on nucleotide chemical properties. Bioinformatics 2017;33(22):3518–23.
[22] He W, Jia C, Zou Q. 4mcpred: machine learning methods for dna n4-methylcytosine sites prediction. Bioinformatics 2019;35(4):593–601.
[23] Wei L, Luan S, Nagai LAE, Su R, Zou Q. Exploring sequence-based features for the improved prediction of dna n4-methylcytosine sites in multiple species. Bioinformatics 2019;35(8):1326–33.
[24] Manavalan B, Basith S, Shin TH, Wei L, Lee G. Meta-4mcpred: a sequence-based meta-predictor for accurate dna 4mc site prediction using effective feature representation. Mol Ther-Nucl Acids 2019;16:733–44.
[25] Zeng F, Fang G, Yao L. A deep neural network for identifying dna n4-methylcytosine sites. Front Genet 2020;11:209.

[26] Xu H, Jia P, Zhao Z. Deep4mc: systematic assessment and computational prediction for dna n4-methylcytosine sites by deep learning. Briefings in Bioinformatics..
[27] Zeng R, Liao M. Developing a multi-layer deep learning based predictive model to identify dna n4-methylcytosine modifications. Front Bioeng Biotechnol 8..
[28] Liu Q, Chen J, Wang Y, Li S, Jia C, Song J, Li F. Deeptorrent: a deep learning-based approach for predicting dna n4-methylcytosine sites. Briefings in Bioinformatics..
[29] Manavalan B, Basith S, Shin TH, Lee DY, Wei L, Lee G, et al. 4mcpred-el: an ensemble learning framework for identification of dna n4-methylcytosine sites in the mouse genome. Cells 2019;8(11):1332.
[30] Wei L, Su R, Luan S, Liao Z, Manavalan B, Zou Q, Shi X. Iterative feature representations improve n4-methylcytosine site prediction. Bioinformatics 2019;35(23):4930–7.
[31] Yang J, Lang K, Zhang G, Fan X, Chen Y, Pian C. Somm4mc: a second-order markov model for dna n4-methylcytosine site prediction in six species. Bioinformatics..
[32] Tang Q, Kang J, Yuan J, Tang H, Li X, Lin H, Huang J, Chen W. Dna4mc-lip: a linear integration method to identify n4-methylcytosine site in multiple species. Bioinformatics 2020;36(11):3327–35.
[33] Hasan MM, Manavalan B, Khatun MS, Kurata H. i4mc-rose, a bioinformatics tool for the identification of dna n4-methylcytosine sites in the rosaceae genome. Int J Biol Macromol 2020;157:752–8.
[34] Wahab A, Mahmoudi O, Kim J, Chong KT. Dnc4mc-deep: identification and analysis of dna n4-methylcytosine sites based on different encoding schemes by using deep learning. Cells 2020;9(8):1756.
[35] Mikolov T, Chen K, Corrado G, Dean J. Efficient estimation of word representations in vector space, arXiv preprint arXiv:1301.3781..
[36] Asgari E, Mofrad MR. Continuous distributed representation of biological sequences for deep proteomics and genomics. PloS one 2015;10(11):e0141287.
[37] Le NQK, Yapp EKY, Ho Q-T, Nagasundaram N, Ou Y-Y, Yeh H-Y. ienhancer-5step: identifying enhancers using hidden information of dna sequences via chou's 5-step rule and word embedding. Anal Biochem 2019;571:53–61.
[38] Khanal J, Tayara H, Chong KT. Identifying enhancers and their strength by the integration of word embedding and convolution neural network. IEEE Access 2020;8:58369–76.
[39] Habibi M, Weber L, Neves M, Wiegandt DL, Leser U. Deep learning with word embeddings improves biomedical named entity recognition. Bioinformatics 2017;33(14):i37–48.
[40] Hamid M-N, Friedberg I. Identifying antimicrobial peptides using word embedding with deep recurrent neural networks. Bioinformatics 2019;35(12):2009–16.
[41] Öztürk H, Ozkirimli E, Özgür A. A novel methodology on distributed representations of proteins using their interacting ligands. Bioinformatics 2018;34(13):i295–303.
[42] Zeng W, Wu M, Jiang R. Prediction of enhancer-promoter interactions via natural language processing. BMC Genomics 2018;19(2):13–22.
[43] Nazari I, Tahir M, Tayara H, Chong KT. in6-methyl (5-step): Identifying rna n6-methyladenosine sites using deep learning mode via chou's 5-step rules and chou's general pseknc. Chemometrics Intell Lab Syst 2019;193:103811.
[44] Hinton G, Deng L, Yu D, Dahl GE, Mohamed A-R, Jaitly N, Senior A, Vanhoucke V, Nguyen P, Sainath TN, et al. Deep neural networks for acoustic modeling in speech recognition: the shared views of four research groups. IEEE Signal Process Mag 2012;29(6):82–97.
[45] Tayara H, Soo KG, Chong KT. Vehicle detection and counting in high-resolution aerial images using convolutional regression neural network. IEEE Access 2017;6:2220–30.
[46] Tayara H, Chong KT. Object detection in very high-resolution aerial images using one-stage densely connected feature pyramid network. Sensors 2018;18(10):3341.
[47] Collobert R, Weston J, Bottou L, Karlen M, Kavukcuoglu K, Kuksa P. Natural language processing (almost) from scratch. J Mach Learn. Res 2011;12(ARTICLE):2493–537.
[48] Zou J, Huss M, Abid A, Mohammadi P, Torkamani A, Telenti A. A primer on deep learning in genomics. Nat Genet 2019;51(1):12–8.
[49] Tayara H, Tahir M, Chong KT. Identification of prokaryotic promoters and their strength by integrating heterogeneous features. Genomics 2020;112(2):1396–403.
[50] Oubounyt M, Louadi Z, Tayara H, Chong KT. Deep learning models based on distributed feature representations for alternative splicing prediction. IEEE Access 2018;6:58826–34.
[51] Alam W, Ali SD, Tayara H, Chong K. A cnn-based rna n6-methyladenosine site predictor for multiple species using heterogeneous features representation. IEEE Access 2020;8:138203–9.
[52] Ng P. dna2vec: consistent vector representations of variable-length k-mers, arXiv preprint arXiv:1701.06279..
[53] Ali SD, Alam W, Tayara H, Chong K. Identification of functional pirnas using a convolutional neural network. IEEE/ACM Trans Comput Biol Bioinf..
[54] Liu Z-Y, Xing J-F, Chen W, Luan M-W, Xie R, Huang J, Xie S-Q, Xiao C-L. Mdr: an integrative dna n6-methyladenine and n4-methylcytosine modification database for rosaceae. Horticulture Res 2019;6(1):1–7.
[55] Fu L, Niu B, Zhu Z, Wu S, Li W. Cd-hit: accelerated for clustering the next-generation sequencing data. Bioinformatics 2012;28(23):3150–2.
[56] Harris ZS. Distributional structure. Word 1954;10(2–3):146–62.
[57] Searls DB. String variable grammar: a logic grammar formalism for the biological language of dna. J Logic Programm 1995;24(1–2):73–102.
[58] Yandell MD, Majoros WH. Genomics and natural language processing. Nat Rev Genet 2002;3(8):601–10.
[59] Meche CE, Hoffmeyer J. From language to nature: the semiotic metaphor in biology..
[60] Cohen KB, Hunter L. Natural language processing and systems biology. In: Artificial intelligence methods and tools for systems biology. Springer; 2004. p. 147–73.
[61] Du J, Jia P, Dai Y, Tao C, Zhao Z, Zhi D. Gene2vec: distributed representation of genes based on co-expression. BMC genomics 2019;20(1):82.
[62] Recalde L, Mendieta J, Boratto L, Terán L, Vaca C, Baquerizo G. Who you should not follow: extracting word embeddings from tweets to identify groups of interest and hijackers in demonstrations. IEEE Trans Emerg Top Comput 2017;7(2):206–17.
[63] Rehurek R, Sojka P. Software framework for topic modelling with large corpora. In: Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks Citeseer.
[64] Hinton G, Srivastava N, Swersky K. Neural networks for machine learning lecture 6a overview of mini-batch gradient descent, Cited on 14 (8)..
[65] De Boer P-T, Kroese DP, Mannor S, Rubinstein RY. A tutorial on the cross-entropy method. Ann Oper Res 2005;134(1):19–67.
[66] Pedregosa F, Varoquaux G, Gramfort A, Michel V, Thirion B, Grisel O, Blondel M, Prettenhofer P, Weiss R, Dubourg V, et al. Scikit-learn: machine learning in python. J Mach Learn Res 2011;12:2825–30.
[67] Khanal J, Lim DY, Tayara H, Chong KT. i6ma-stack: a stacking ensemble-based computational prediction of dna n6-methyladenine (6ma) sites in the rosaceae genome. Genomics 2021;113(1):582–92.
[68] Siraj A, Chantsalnyam T, Tayara H, Chong KT. Recsno: prediction of protein s-nitrosylation sites using a recurrent neural network. IEEE Access 9: 6674–6682..
[69] Lim DY, Khanal J, Tayara H, Chong KT. ienhancer-rf: identifying enhancers and their strength by enhanced feature representation using random forest. Chemometrics Intell Lab Syst 2021;104284.
[70] Bradley AP. The use of the area under the roc curve in the evaluation of machine learning algorithms. Pattern Recogn 1997;30(7):1145–59.
[71] Fernández A, García S, Galar M, Prati RC, Krawczyk B, Herrera F. Learning from imbalanced data sets. Springer; 2018.
[72] Vacic V, Iakoucheva LM, Radivojac P. Two sample logo: a graphical representation of the differences between two sets of sequence alignments. Bioinformatics 2006;22(12):1536–7.
[73] Johnson JM, Khoshgoftaar TM. Survey on deep learning with class imbalance. J Big Data 2019;6(1):27.
[74] Saito T, Rehmsmeier M. The precision-recall plot is more informative than the roc plot when evaluating binary classifiers on imbalanced datasets. PloS one 2015;10(3):e0118432.
[75] He H, Ma Y. Imbalanced learning: foundations, algorithms, and applications. John Wiley & Sons; 2013.