XGB4mcPred: Identification of DNA N4-Methylcytosine Sites in Multiple Species Based on an eXtreme Gradient Boosting Algorithm and DNA Sequence Information

Xiao Wang 1*, Xi Lin 1, Rong Wang 1, Kai-Qi Fan 2, Li-Jun Han 1 and Zhao-Yuan Ding 1

1 School of Computer and Communication Engineering, Zhengzhou University of Light Industry, Zhengzhou 450002, China; linxi@zzuli.edu.cn (X.L.); wangrong@zzuli.edu.cn (R.W.); hanlijun@zzuli.edu.cn (L.-J.H.); dingzhao@zzuli.edu.cn (Z.-Y.D.)
2 School of Material and Chemical Engineering, Zhengzhou University of Light Industry, Zhengzhou 450002, China; fankaqi@zzuli.edu.cn

*Correspondence: wangxiao@zzuli.edu.cn

Abstract: DNA N4-methylcytosine(4mC) plays an important role in numerous biological functions and is a mechanism of particular epigenetic importance. Therefore, accurate identification of the 4mC sites in DNA sequences is necessary to understand the functional mechanism. Although some effective calculation tools have been proposed to identifying DNA 4mC sites, it is still challenging to improve identification accuracy and generalization ability. Therefore, there is a great need to build a computational tool to accurately identify the position of DNA 4mC sites. Hence, this study proposed a novel predictor XGB4mcPred, a predictor for the identification of 4mC sites trained using an extreme gradient boosting algorithm (XGBoost) and DNA sequence information. Firstly, we used the One-Hot encoding on adjacent and spaced nucleotides, dinucleotides, and trinucleotides of the original 4mC site sequences as feature vectors. Then, the importance values of the feature vectors pre-trained by the XGBoost algorithm were used as a threshold to filter redundant features, resulting in a significant improvement in the identification accuracy of the constructed XGB4mcPred predictor to identify 4mC sites. The analysis shows that there is a clear preference for nucleotide sequences between 4mC sites and non-4mC site sequences in six datasets from multiple species, and the optimized features can better distinguish 4mC sites from non-4mC sites. The experimental results of cross-validation and independent tests from six different species show that our proposed predictor XGB4mcPred significantly outperformed other state-of-the-art predictors and was improved to varying degrees compared with other state-of-the-art predictors. Additionally, the user-friendly webserver we used to developed the XGB4mcPred predictor was made freely accessible.

Keywords: N4-methylcytosine; eXtreme gradient boosting; site identification; feature selection

1. Introduction

DNA methylation is the process of adding methyl groups to specific regions of DNA and leads to genetic changes in gene expression [1], and this process can regulate gene expression and shutdown without altering the nucleotide sequence. Therefore, the activity of genes can be regulated by controlling the process of DNA methylation, thus turning off certain gene activity or inducing the reactivation and expression of certain genes [2–5]. Hence, it is closely relevant for the study of cancer, aging, or for the study of regulating virulence and antibiotic resistance in prokaryotes [6–8]. Each DNA sequence is composed of multiple nucleotide bases, and there are four types of bases: adenine (A), thymine (T), cytosine (C), and guanine (G). Depending on where the methylation group binds to a base in the DNA sequences, DNA base methylation can be divided into various types: N6-methyladenine (6mA), which occurs at the N-6 position of adenine; N7-methylguanosine (7mG), which occurs at the N-7 position of guanine; 5-methylcytosine (5mC) and N4-methylcytosine (4mC), which occur at the C-5 and C-4 position of cytosine, respectively; and
so on. Among them, the DNA 5mC sites, 6mA sites, and 4mC sites are the most common methylation processes, which are widely found in eukaryotes and prokaryotes [9–12]. Compared with the studies for 5mC and 6mA, the detection of 4mC is relatively difficult due to the lack of adequate assays. Additionally, as the 4mC modification site is a member of the restriction endonuclease modification (RM) system, which not only protects host DNA from degradation by restriction enzymes [13], but also controls DNA replication and cell cycle and corrects DNA replication errors, among other functions [14], the study of accurate identification of the 4mC site is crucial. Some wet experiment techniques have been proposed to accurately identify these 4mC sites, for example, single-molecule real-time sequencing technology (SMRT) for identifying DNA 4mC modification sites, and the 4mC-TAB-seq method that can accurately identify 4mC modification sites without interference from 5mC modification [15,16]. However, the identification of 4mC sites through these wet experimental techniques is very time consuming and costly. Therefore, the development of bioinformatics tools for the accurate and efficient identification of 4mC sites by computational methods is a useful complement to the experimental method.

iDNA4mC [17] is the first prediction tool proposed by Wei Chen and Hao Lin to predict 4mC modification sites using machine learning methods. The positive samples containing 4mC sites were obtained from the MethSMRT database [18], involving six species of *Escherichia coli*, *Geobacter pickeringii*, *Geoalkalibacter subterraneus*, *Caenorhabditis elegans*, *Arabidopsis thaliana*, and *Drosophila melanogaster*, and finally, a high-quality balanced dataset was constructed using the sliding window method (the optimal window length is 41 bp). Subsequently, based on the benchmark dataset of iDNA4mC predictors, Zou Quan’s team used electron ion interaction potential (EIIP) and position-specific trinucleotide propensity (PSTNP) feature expression methods to propose the second predictor 4mCPred [19] to identify 4mC modification sites. The results showed that, compared to the iDNA4mC predictor, the predictor’s overall performance had improved. Based on the first two predictors, Zou Quan and his team proposed the 4mcPred-SVM [20] predictor based on the feature representation method of sequence information, in which the feature selection optimization strategy was used to screen out the optimal model corresponding to the model. Feature subsets were used to obtain representative features, and then SVM and the obtained features were used for supervised training to obtain optimal models for different species, and the effect was improved. Later, Manavalan et al. adopted a feature representation learning scheme. A total of 56 probabilistic features were generated based on seven different feature representations and four different machine learning algorithms, and then probabilistic features were used as the input of the support vector machine to finally establish a Meta-4mCpred predictor [21]. After that, Leyi Wei proposed a predictor 4mcPred-IFL [22] that used an iterative feature representation method based on composition information, physicochemical information, and location-specific information, which aimed to identify 4mC-modified sites with higher accuracy. Recently, several 4mC site prediction methods based on deep learning techniques have been developed with decent enhancements. Khanal et al. constructed a 4mCCNN [23] predictor using end-to-end convolutional neural networks. Xu et al. applied convolutional neural networks with four representative features to construct the Deep4mC [24] predictor. Liu et al. used a multi-layer convolutional and pooling layer to extract features, followed by the Attention mechanism, LSTM, and fully connected layers for training; finally, they constructed the DeepTorrent [25] predictor, which achieved the top results. Although the final results of these recent papers have well improved on the benchmark datasets, the complexity of the models has also greatly improved, which causes the predictor’s robustness to decrease along with other characteristics, and in some independent tests, the identifying effect on the set is always unsatisfactory. Therefore, it is essential to develop a simple, effective, and robust predictor.

This study proposed a novel predictor named XGB4mcPred based on a One-Hot encoding genomic sequence information. The framework of the proposed XGB4mcPred predictor is shown in Figure 1. To extract more information features, this study encoded
different types of DNA sequence information, including adjacent information and spaced information of different spaced sizes. Afterward, the extracted genomic sequence information optimized through a two-step feature optimization strategy to obtain a DNA sequence information representation of the 4mC sites. The first step was to use the XGBoost classifier to pre-train the One-Hot encoded feature vectors to obtain the feature importance values of the number of feature dimensions. After that, in the second step, all the calculated feature importance values were deduplicated and sorted from small to large, and then non-repetitive feature importance values were used as the threshold for feature selection to perform feature filtering in order to determine the optimal DNA sequence information for each predictor. The filtering evaluation method used the prediction accuracy of the default XGBoost classifier. After that, the four machine learning algorithms of KNN, SVM, RF, and XGBoost were used to supervise training and hyperparameter adjustment on the determined optimal DNA sequence information, and finally, a novel predictor with better performance was constructed based on the XGBoost learning algorithm. After that, two different feature fusion strategies, parallel fusion and serial fusion, were compared. Through comparative analysis, it was concluded that using a serial fusion strategy can make genome sequence information representation more effective. Therefore, it was finally determined that using the feature fusion strategy of serial fusion, the predictor obtained by training the selected optimal feature subset on the XGBoost machine learning algorithm had the best accuracy. A user-friendly webserver where the XGB4mcPred predictor was developed can be accessed for free at http://www.xwanglab.com/XGB4mcPred/ (accessed on 19 July 2021).
Figure 1. A framework of DNA 4mC site prediction based on the XGB4mcPred predictor.

2. Materials and Methods

2.1. Datasets

This study adopted six different species benchmark datasets and independent datasets used in iDNA4mC by Chen et al. [17]. The six different species are *Escherichia coli* (*E. coli*), *Geobacter pickeringii* (*G. pickeringii*), *Geoalkalibacter subterraneus* (*G. subterraneus*), *Caenorhabditis elegans* (*C. elegans*), *Arabidopsis thaliana* (*A. thaliana*), and *Drosophila melanogaster* (*D. melanogaster*). Additionally, 776, 1138, 1852, 3108, 3956, and 3538 are the number of sequences in each species, respectively. Furthermore, they have the same number of 4mC site sequences and non-4mC site sequences. Preliminary experiments in previous studies have shown that the best predictions effect when the sequence length is 41 bp. Therefore, the DNA sequence length of all 4mC site sequences and non-4mC site sequences in the samples in the six datasets is 41 bp. To provide a comprehensive and unbiased comparison, we also used the six independent datasets constructed using the same methodology in the Meta-4McPred. Finally, they obtained 134, 200, 350, 750, 1250, and 1000 number of 4mC sites, respectively, from the *E. coli*, *G. pickeringii*, *G. subterraneus*, *C. elegans*, *A. thaliana*, and *D. melanogaster* genomes. The details of how those datasets were constructed can be re-
yielded very effective results, especially the random forest algorithm. The random forest
Boosting is used to obtain new classifiers by focusing on the DNA sequence information
by the Boosting and Bagging algorithms are the same. Unlike the Bagging technique,
output, which is likely to avoid the bad sample data and, thus, improve the accuracy.
training. After many times with these results, the average value is obtained as the resulting
2.3. eXtreme Gradient Boosting Algorithm (XGBoost)
2.2. Sequence Encoding
One-Hot encoding encodes the original DNA sequence directly, thus retaining the
for DNA sequences. It can be formulated as Equation (3).
\[ R_i, \]
\[ R_iR_{i+(gap_k+1)}, \]
\[ R_iR_{i+(gap_k+1)}R_{i+2(gap_k+1)}, \]
\[ \ldots \]
\[ R_iR_{i+(gap_k+1)}R_{i+2(gap_k+1)} \ldots R_{i+(m-1)(gap_k+1)}, \]
\[ k = 1, \; gap_k = 0, 1, 2 \ldots \]
\[ k = 2, \; gap_k = 0, 1, 2 \ldots \]
\[ k = 3, \; gap_k = 0, 1, 2 \ldots \]
\[ \ldots \]
\[ k = m, \; gap_k = 0, 1, 2 \ldots \]

We constructed a preliminary experimental screening in advance, and the experiment
finally determined that when \( k = 1, \; gap_k = 0; k = 2, \; gap_k = 0; k = 2, \; gap_k = 1; \)
\( k = 2, \; gap_k = 2; \) and \( k = 2, \; gap_k = 3 \) were combined, the feature effect was the best.
Hence, the sequence of DNA can be represented as \( (4 + L + 16 \times (L - 1) + 16 \times (L - 2) + 16 \times (L - 3) + 16 \times (L - 4)) \)-dimensional vectors. In the six datasets, the sequence length is
41 bp, so each DNA sequence eventually contains vectors of 2628 dimensions.

2.3. eXtreme Gradient Boosting Algorithm (XGBoost)
Tree ensemble classifiers have been widely used in the field of bioinformatics and have
yielded very effective results, especially the random forest algorithm. The random forest
algorithm uses the Bagging method to fuse each decision tree, and Bagging is bootstrap aggregating. The idea is to randomly select a part of the samples from the population for training. After many times with these results, the average value is obtained as the resulting output, which is likely to avoid the bad sample data and, thus, improve the accuracy.

However, in this study, another method was trained in our predictor, i.e., Boosting. Boosting is a technique similar to Bagging. The basic classifier types that are used internally by the Boosting and Bagging algorithms are the same. Unlike the Bagging technique, Boosting is used to obtain new classifiers by focusing on the DNA sequence information
of 4mC sites that are misclassified by existing classifiers. Here, we chose the extremely famous library for the boosted tree learning model, XGBoost. It is a gradient boosting implementation of a highly efficient system. The XGBoost not only uses the Taylor expansion quadratic approximation of the objective function during iterative optimization to speed up the optimization process, but also adds regularization terms to the objective function to effectively prevent overfitting. For further information about the algorithm, refer to [26].

For XGBoost, the hyperparameter used in this study for optimization includes the following: max_depth, gamma, learning_rate, subsample, colsample_bytree, colsample_bylevel, reg_alpha, and reg_lambda. All possible combinations of these parameter values were experimented with many times, and the parameter values that made the model perform best were kept as optimization values. Considering the size and characteristic dimensions of those datasets, the hyperparameter setting range of the XGBoost classifier in this study is shown in Table 1.

### Table 1. The hyperparameters of the XGBoost classifier.

| Hyperparameter         | Range          |
|------------------------|----------------|
| max_depth              | 3–7            |
| learning_rate          | 0.01–0.2       |
| colsample_bytree       | 0.3–1          |
| subsample              | 0.3–1          |
| gamma                  | 0–1.5          |
| colsample_bylevel      | 0.5–0.9        |
| reg_alpha              | 1–1.5          |
| reg_lambda             | 0–1.5          |

2.4. Model Evaluation

In the field of bioinformatics and recent studies, four metrics are used to evaluate the quality of predictors. They are Specificity (SP), Sensitivity (SN), Accuracy (ACC), and Mathew’s correlation coefficient (MCC). Their formulas are as follows:

\[
SN = \frac{TP}{TP + FN} \tag{4}
\]

\[
SP = \frac{TN}{TN + FP} \tag{5}
\]

\[
ACC = \frac{TP + TN}{TP + TN + FN + FP} \tag{6}
\]

\[
MCC = \frac{TP \times TN - FP \times FN}{\sqrt{(TP + FN)(TP + FP)(TN + FP)(TN + FN)}} \tag{7}
\]

where TP (true positive) represents observation in 4mC sites and is predicted to be the 4mC sites; TN (true negative) represents observation in non-4mC sites and is predicted to be the non-4mC sites; FN (false negative) represents observation in 4mC sites but is predicted to be negative; FP (false positive) represents observation in non-4mC sites but is predicted to be 4mC sites.

Hence, SN is the probability of getting the prediction right in the 4mC sites. SP is the probability of getting the prediction right in the non-4mC sites. ACC represents the accuracy of the prediction of the overall sites. Because the MCC takes into account true positive, false positive, true negative, and false negative, it is often seen as a measure of balance.

3. Results and Discussion

In this section, this study first performed a nucleotide composition preference analysis for DNA 4mC site sequences and non-4mC site sequences to verify the effectiveness of One-Hot encoding for multiple types of sequences as proposed in this study, followed
by a detailed description of how to further improve the representation of DNA sequence information for the extracted features and comparative analysis. After that, not only was the performance between different machine learning algorithms compared, but also, a comparative analysis of two different feature fusion strategies was performed. Finally, the predictor XGBmcPred constructed in this study was compared to current state-of-the-art predictors on benchmark datasets and independent datasets.

3.1. Nucleotide Composition Analysis

To explore why the coding approach proposed in this study worked effectively, we used the Two Sample Logos software [27] on all six species datasets to analyze the position-specific preferences of the nucleotide composition of these species as a way to explore statistically significant nucleotide differences between sequences containing non-4mC sites and 4mC sites. Figure 2 shows the nucleotide preferences of 41 base pairs of DNA sequences from three-scale datasets, where the C base pair is centrally positioned. Information on nucleotide preferences for the other three datasets is referenced in the provided Supplementary Figure S1. As shown in Figure 2, E. coli and G. subterraneus adenosine and thymine are significantly enriched on sequences containing 4mC sites, while sequences at non-4mC sites exhibit significant cytosine and guanine preferences. In contrast, cytosine and guanine are significantly enriched on sequences containing 4mC sites and show a clear preference for adenosine and thymine on non-4mC site sequences on D. melanogaster.

Figure 2. Sequence logos between 4mC sites and non-4mC sites of Escherichia coli, Geoalkalibacter subterraneus, and Drosophila melanogaster.

Taking a detailed example, at position 16 in the E. coli dataset, adenosine and thymine are particularly significant on sequences containing the 4mC site, while at the same position, cytosine and guanine have a significant preference on sequences of the non-4mC site. Similarly, at position 24, guanine and thymine are particularly significant on sequences containing the 4mC site, and cytosine and adenosine are particularly significant on sequences
in the non-4mC site. Additionally, at positions 19, 20, 22, and 27, there are also significant differences between the samples containing 4mC sites and non-4mC sites.

However, at positions 9, 10, 11, 12, 35, 36, and 37, there are no significant differences between the 4mC sites and non-4mC sites. Therefore, we considered coding for consecutive positions or spaced positions for dinucleotides or trinucleotides as a way to highlight the differences between these positions. For example, assuming we encode dinucleotides spaced three nucleotide residues apart, then, when we encode position 12, we are encoding both positions 12 and 16, and at position 16 there is a clear nucleotide preference between the 4mC site sequences and the non-4mc site sequences at this time. Therefore, this information will assist in the differentiation of nucleotides preferences at position 12. One-Hot encoding is the direct coding of the original nucleotide residues and, therefore, contains the most original nucleotide sequence information. The distinction of the 4mC sites is further enhanced by combining consecutive, spaced nucleotide residues to assist in boosting the nucleotide preference at a particular position, which would be very beneficial in improving the accuracy of 4mC site recognition.

3.2. Determine the Optimal Features Spaces

The optimal features were selected from the six species using the Wrapper method. To determine optimal feature spaces, the XGBoost algorithm was first used to rank the features according to their importance values from minimum to maximum. Due to the fact that genomic sequence information encoding produces a large number of redundant and discrete features, it causes generous duplicate importance values. Hence, secondly, we regarded the feature importance values as thresholds for feature selection, and features with an importance value less than the threshold were discarded. The subset of features was continuously selected from the original features by thresholds, and it was evaluated according to the performance of XGBoost. As shown in Figure 3, the accuracy of 10-fold cross-validation varied as the features’ threshold ranking was added in six species. Take the E.coli dataset, for example, when the threshold ranking was 67, the performance of XGBoost was the highest when the shape corresponded to the (776, 194); in other words, when the threshold ranking was 67, the feature dimension was reduced to 194, and the XGBoost had the highest performance in the E. coli dataset of 776 samples.

Figure 3. Accuracies of the proposed features on datasets from Caenorhabditis elegans, Drosophila melanogaster, Arabidopsis thaliana, Escherichia coli, Geoalkalibacter subterraneus, and Geobacter pickeringii as the number of selected features increases.
This study compared the performance of the optimal features (after feature selection) and the original features (before feature selection) to verify whether the proposed feature selection strategy could improve the prediction performance. These two features were trained on the same dataset by the XGBoost algorithm and then evaluated using 10-fold cross-validation. Table 2 shows results of all the metrics (ACC, MCC, SP, SN). In all six species, the optimal features outweigh the original features to varying degrees. The results show that an average accuracy improvement of 1.7–4.2% was achieved using feature optimization compared to the original features; the MCC improved by an average of 2.7–9.8%. In Table 2, one can also intuitively see that the feature size of the six species was greatly reduced, especially in A. thaliana, where the feature size was reduced from 2628 dimensions to 184 dimensions. Again, a visual comparison was made between the original features and the optimal features, as shown in Figure 4.

| Species          | Features | ACC   | MCC   | SN    | SP    | Size |
|------------------|----------|-------|-------|-------|-------|------|
| C. elegans       | original | 0.844 | 0.689 | 0.852 | 0.837 | 2628 |
|                  | optimal  | 0.859 | 0.718 | 0.866 | 0.853 | 214  |
| D. melanogaster  | original | 0.846 | 0.692 | 0.864 | 0.828 | 2628 |
|                  | optimal  | 0.859 | 0.719 | 0.875 | 0.844 | 228  |
| A. thaliana      | original | 0.787 | 0.574 | 0.784 | 0.790 | 2628 |
|                  | optimal  | 0.804 | 0.609 | 0.799 | 0.809 | 184  |
| E. coli          | original | 0.854 | 0.709 | 0.874 | 0.835 | 2628 |
|                  | optimal  | 0.903 | 0.807 | 0.922 | 0.884 | 194  |
| G. subterreneus  | original | 0.840 | 0.681 | 0.832 | 0.847 | 2628 |
|                  | optimal  | 0.864 | 0.729 | 0.846 | 0.882 | 241  |
| G. pickeringii   | original | 0.867 | 0.735 | 0.858 | 0.877 | 2628 |
|                  | optimal  | 0.902 | 0.805 | 0.888 | 0.917 | 232  |

Figure 4. Performance of the original features and optimal features on the Accuracy and Matthews correlation coefficient metrics.

The optimized feature space described above is proven to effectively improve performance. However, how the learned features contribute to performance improvement is unknown. Here, this study further explored the change from the original feature space to the optimal feature space distribution to understand how features contribute to performance improvement. Hence, this study used T-distributed Stochastic Neighbor Embedding (t-SNE) technology [28] to reduce the dimensionality of the feature space to visualize the feature space. Figure 5 shows the three-scale datasets before and after determining the optimal features. We can observe the difference between the original features and the optimized features. Before optimizing features, the distribution between positive and negative samples is relatively scattered, but after optimizing features, the two clusters of
positive and negative samples are significantly more clustered. This result makes it easier for the predictor to place positive and negative samples in the same category in the process of drawing decision boundaries, thus improving classification accuracy. It indicates that the genomic sequence information after determining the optimized features may somehow better distinguish between non-4mC sites and 4mC sites. The information of the other species is shown in Supplementary Figure S2.

3.3. XGBoost Comparison with Other ML Algorithms

We compared the predictive performance of the XGBoost with other machine learning algorithms, including K-Nearest Neighbors (KNN), RandomForest (RF), and Support Vector Machine (SVM) algorithms. For comparison, the three machine learning algorithms described above replaced XGBoost in the predictor and retrained the prediction predictor, which was then evaluated by 10-fold cross-validation on the six datasets. The performances of the four models are depicted in Figure 6. It indicates the performances in terms of the four metrics (SP, SN, ACC, MCC). It can be observed that, for the two algorithms of KNN and RF, both ACC and MCC are far inferior to XGBoost in the six species, and on the metrics of SN and SP, the KNN algorithm is very unstable and has large deviations. Taking E. coli as an example, the SN metric of KNN reached the highest value of 95.4%, while the SP metric was very low at only 74.7%. On the contrary, in the G. pickeringii, the SP metric of KNN had a higher level of 91.9%, but the SN metric was the lowest among the four ML algorithms and does not exceed 80%. However, for the SVM of the C. elegans, D. melanogaster, and G. subterraneus species, the performance of SVM and XGBoost were almost the same, the MCC value of D. melanogaster and G. subterraneus on XGBoost was only 0.2% higher than SVM. For the other three species, the ACC values of the XGBoost models for A. thaliana, E. coli, and G. pickeringii outperformed the corresponding SVM models by 1.1%, 1.4%, and 1.5%, respectively, and the MCC for the XGBoost improved by an average of 1.1–3.2%. Therefore, the overall performance of the four classifiers, the XGBoost was the best. Hence, the XGBoost was used to establish computational tools in the following.
3.4. Comparative Analysis on Two Feature Fusion Strategies

In this section, to further evaluate the optimal genomic sequence information determined above, we comparatively analyze the effects of two fusion strategies, serial fusion and parallel fusion. The serial fusion strategy means that the information of the five genomic sequences represented by the One-Hot encoding is directly serially merged, and later the features are screened using the two-step feature selection method described above as the optimal genomic sequence information representation for the 4mC sites. The parallel fusion strategy means that the five genomic sequences’ information, represented by One-Hot encoding, is respectively screened for each sequence using the two-step feature selection method described above, and then a subset of these five genomic sequences’ information is combined and represented as the optimal genomic sequence information for the 4mC sites.

Figure 7 shows the comparative results of both ACC and MCC metrics for these two fusion strategies on the six species. Specifically, on the C. elegans, D. melanogaster, and G. subterruneus datasets, the results of the parallel and serial fusion strategies were not very different, and overall, the serial fusion strategy slightly outperformed the refrigerator fusion strategy, with MCC values improving by 1.4%, 1.5%, and 1.4%, respectively. Whereas on the E. coli dataset, the serial fusion strategy was substantially better than the parallel fusion strategy with a 6.7% higher value on MCC metric. Additionally, more detailed information can be obtained in Table 3. The results showed that using serial fusion strategy achieved an average 0.7–3.3% improvement in ACC as compared to the parallel fusion strategy, and the MCC improved by an average of 1.4–6.7%. Additionally, Figure 7 intuitively shows that the two metrics of the serial fusion strategy outperform the two metrics of the parallel fusion strategy on all six species datasets. We speculate that, since the parallel fusion strategy is a separate feature screening of these several genomic sequences’ information, there may still be some duplicate and redundant sequence information in the screened subsets of these genomic sequences, which affects the recognition performance of the final predictor. While using the serial fusion strategy is a direct serial merging of them, the feature screening process is a search in the full space, the search is relatively more thorough, and the final selected genomic sequence information representation will be relatively more precise; therefore, the identification performance of the final predictor will be more accurate.
At the same time, we also conducted a significance test of the final prediction results of parallel fusion and serial fusion. The test criterion was that the prediction accuracy was calculated by 10% of the DNA sequences randomly selected from independent datasets and the operation was repeated 50 times for each species. The significance test was performed from the prediction accuracy for both models. Therefore, for the variance test, the test hypothesis is that there is no significant difference between the prediction accuracy of the serial fusion model and the parallel fusion model.

Through calculation, we concluded that the \( p \)-value of each species is less than 2.5% in the six species. Among them, the \( D. \ melanogaster \) species and \( G. \ subterraneus \) species, which had slightly improved accuracy values, had \( p \)-values that also reached 0.0207 and 0.0223, respectively. Therefore, it can be explained that the original hypothesis is not valid for the six species. The details are shown in Table 3. Therefore, as shown in the table, it can be explained that in \( C. \ elegans \) species, \( D. \ melanogaster \) species, and \( G. \ subterraneus \) species, the prediction results of the serial fusion strategy had a 97.5% probability of significant improvement over the parallel fusion strategy. In the \( A. \ thaliana \) species, \( E. \ coli \) species, and \( G. \ pickeringii \) species, the prediction results of the serial fusion strategy had a 99% probability of significantly improvement over the parallel fusion strategy.

### 3.5. Comparison with State-of-the-Art Predictors

To demonstrate the superiority of our predictor XGB4mcPred for identifying 4mC sites, we compared its performance with that of other predictors, such as iDNA4mC [17], 4mCPred [19], 4mcPred-SVM [20], Meta-4mCpred [21], and the predictor constructed using deep learning techniques 4mCCNN [23], as they were all developed using the same benchmark datasets. Supplementary Table S1 compares the 10-fold cross-validation performance of the state-of-the-art 4mC site predictor and the XGB4mcPred. Compared
to traditional machine learning methods, the results show that the ACC and MCC of the proposed predictor XGB4mcPred on the datasets of six species exceeded the current state-of-the-art predictor to varying degrees. Especially on the E. coli dataset, the ACC of the XGB4mcPred predictor was 9.5% higher than the first predictor iDNA4mC and 4.4% higher than the latest Meta-4mCpred. For the A. thaliana dataset and the G. pickeringii datasets, the 10-fold cross-validation ACC also exceeded 80% and 90% for the first time. Similarly, the value of MCC increased by 1.8% at the lowest and 8.7% at the highest on the datasets of the six species datasets. Additionally, in comparison with the currently popular deep learning methods, the 10-fold cross-validation results of the XGB4mcPred predictor also show varying degrees of improvement. In particular, for the E. coli species, our predictor improved accuracy by 3.3% and obtained a significant improvement in MCC values of 9.6% compared to 4mCCNN. In contrast, on D. melanogaster species, our predictor performed almost as well as 4mCCNN, outperforming it by only 0.1%. Overall, it appears that our predictor did obtain a good improvement in the 10-fold cross-validation, and Figure 8 visualizes the results for the four evaluated metrics. Additionally, compared with the previous complex ensemble model architectures, our study used a simpler feature encoding method and feature selection strategy. The proposed XGB4mcPred predictor can more accurately identify and predict the 4mC sites or non-4mC sites. Therefore, we hope that our proposed model architecture can become an effective predictive tool for identifying non-4mC sites or 4mC sites.

![Figure 8](image-url)

Figure 8. Performances of the proposed XGB4mcPred and five state-of-the-art predictors on six benchmark datasets from different species.

### 3.6. Validation of Various Methods on Independent Datasets

To further demonstrate the generalization ability of our predictor for identifying 4mC sites, we validated them on the independent datasets built by Manavalan et al. Through applying the XGB4mcPred predictor to the six independent datasets, the four metrics were calculated. The detailed comparison results can be found in Table 4. Since many of the previous predictor webservers are not accessible, we compared it with the six state-of-the-art predictors, including 4mCPred [19], 4mcPred-SVM [20], Meta-4mCpred [21], 4mcPred-IPL [22], Deep4mC [24], and DeepTorrent [25]. Among them, the DeepTorrent is the most advanced predictor available using deep learning techniques. The detailed...
comparison results can be found in Table 4. As shown in Table 3, compared with the previous state-of-the-art predictor built by the traditional machine learning methods, our test results show that our proposed XGB4mcPred predictor outperformed the previous state-of-the-art machine learning predictors on the four datasets of *A. thaliana*, *C. elegans*, *E. coli*, *D. melanogaster*. Especially on the *C. elegans*, the ACC value of our independent test set breaks through to the value of 90% for the first time. Compared to current state-of-the-art machine learning predictors, the accuracy of the *D. melanogaster* dataset also reached 91.8%, which was 1.1~3.2% increased. On the *E. coli* dataset, our predictor significantly improved by 3.7~7.8% compared with the previous state-of-the-art machine learning predictors. For the *A. thaliana* dataset, we also got a slight improvement. However, on the *G. subterruneus* and *G. pickeringii* datasets, our model’s generalization was slightly inferior to the Meta-4mCpred predictor. However, compared with other state-of-the-art machine learning predictors, our predictors are still better than them.

### Table 4. Performances of the proposed XGB4mcPred and five state-of-the-art predictors on independent datasets from six species.

| Species          | Independent Predictors | ACC  | MCC  | SN   | SP   |
|------------------|------------------------|------|------|------|------|
| *C. elegans*     | 4mCPred                | 0.865| 0.731| 0.883| 0.849|
|                  | 4mcPred-SVM            | 0.842| 0.684| 0.828| 0.856|
|                  | Meta-4mCpred           | 0.870| 0.741| 0.843| 0.897|
|                  | 4mcPred-IFL            | 0.815| 0.636| 0.751| 0.88 |
|                  | Deep4mC                | 0.915| 0.832| 0.907| 0.925|
|                  | Deep Torrent           | 0.935| 0.871| 0.936| 0.935|
|                  | XGB4mcPred             | 0.900| 0.800| 0.892| 0.908|
| *D. melanogaster*| 4mCPred                | 0.900| 0.803| 0.933| 0.868|
|                  | 4mcPred-SVM            | 0.886| 0.771| 0.886| 0.885|
|                  | Meta-4mCpred           | 0.906| 0.812| 0.913| 0.899|
|                  | 4mcPred-IFL            | 0.907| 0.814| 0.912| 0.902|
|                  | Deep4mC                | 0.911| 0.822| 0.933| 0.890|
|                  | Deep Torrent           | 0.911| 0.829| 0.972| 0.863|
|                  | XGB4mcPred             | 0.918| 0.836| 0.926| 0.910|
| *A. thaliana*    | 4mCPred                | 0.816| 0.632| 0.842| 0.789|
|                  | 4mcPred-SVM            | 0.824| 0.649| 0.842| 0.806|
|                  | Meta-4mCpred           | 0.855| 0.711| 0.876| 0.834|
|                  | 4mcPred-IFL            | 0.823| 0.650| 0.826| 0.824|
|                  | Deep4mC                | 0.868| 0.739| 0.824| 0.912|
|                  | Deep Torrent           | 0.886| 0.773| 0.902| 0.872|
|                  | XGB4mcPred             | 0.858| 0.718| 0.870| 0.843|
| *E. coli*        | 4mCPred                | 0.817| 0.634| 0.851| 0.784|
|                  | 4mcPred-SVM            | 0.784| 0.569| 0.746| 0.821|
|                  | Meta-4mCpred           | 0.825| 0.650| 0.806| 0.843|
|                  | 4mcPred-IFL            | 0.795| 0.594| 0.731| 0.858|
|                  | Deep4mC                | 0.754| 0.517| 0.815| 0.713|
|                  | Deep Torrent           | 0.818| 0.634| 0.810| 0.824|
|                  | XGB4mcPred             | 0.862| 0.726| 0.821| 0.903|
| *G. subterruneus*| 4mCPred                | 0.789| 0.578| 0.757| 0.820|
|                  | 4mcPred-SVM            | 0.811| 0.624| 0.783| 0.840|
|                  | Meta-4mCpred           | 0.850| 0.701| 0.817| 0.883|
|                  | 4mcPred-IFL            | 0.686| 0.378| 0.591| 0.78 |
|                  | Deep4mC                | 0.830| 0.664| 0.797| 0.871|
|                  | Deep Torrent           | 0.823| 0.646| 0.832| 0.814|
|                  | XGB4mcPred             | 0.814| 0.629| 0.789| 0.840|
| *G. pickeringii*| 4mCPred                | 0.742| 0.503| 0.610| 0.875|
|                  | 4mcPred-SVM            | 0.758| 0.515| 0.750| 0.765|
|                  | Meta-4mCpred           | 0.850| 0.700| 0.835| 0.865|
|                  | 4mcPred-IFL            | 0.678| 0.358| 0.740| 0.615|
|                  | Deep4mC                | 0.803| 0.611| 0.850| 0.767|
|                  | Deep Torrent           | 0.860| 0.735| 0.950| 0.800|
|                  | XGB4mcPred             | 0.820| 0.641| 0.790| 0.850|
Compared to the top predictors currently built through the deep learning technique Deep4mC [24] and DeepTorrent [25] predictors, our generalization ability exceeded the Deep4mC predictor on three species D. melanogaster, E. coli, and G. pickeringii, with a maximum improvement of 10.8% accuracy. Additionally, compared to the DeepTorrent predictor, our predictor slightly exceeded on the E. coli and D. melanogaster species, by 4.4% and 0.7%. However, on the other species, our predictors did not outperform the current top predictors Deep4mC and DeepTorrent, lagging in the range of 1% to 4%. Therefore, we finally calculated the average accuracy values of the three predictors (XGB4mcPred, Deep4mC, and DeepTorrent) on six species to evaluate the differences. The results show that the average accuracy of XGB4mcPred is 0.862, the average accuracy of Deep4mC is 0.847, and the average accuracy of DeepTorrent is 0.872. It can be found that the average accuracy of XGB4mcPred is slightly better than the Deep4mC predictor by 1.5%, and the average accuracy of DeepTorrent is slightly better than the XGB4mcPred predictor by 1%.

Finally, we also compared the AUC values of XGB4mcPred with other state-of-the-art predictors on independent datasets of six species. Figure 9 shows the results of the AUC comparison of the XGB4mcPred predictor with the currently available top predictors. As shown in Figure 9, on the independent dataset of D. melanogaster species, the AUC value of the XGB4mcPred implementation gains a slight boost compared to the other four state-of-the-art predictors, reaching a maximum of 97%, but is 1% less compared to the DeepTorrent predictor. For the independent datasets of C. elegans, A. thaliana, and G. subterruneus species, the XGB4mcPred predictor is above or on par with the 4mcPred-SVM, Meta-4mCpred, and 4mcPred-IFL predictors, but the AUC performance is slightly below that of the Deep4mC and DeepTorrent predictors. For the independent dataset of G. pickeringii species, our predictor reaches the same level as the two most advanced predictors currently available, Meta-4mCpred and Deep4mC, but with 9% less AUC values compared to the DeepTorrent predictor. In contrast, for the independent dataset of E. coli species, the AUC value of our predictor is 9% higher than the Deep4mC predictor and 2% higher than the DeepTorrent predictor.

Figure 9. Performance comparison of XGB4mcPred and other state-of-the-art predictors of the AUC value on independent datasets from multiple species.
We attribute that there are two potential reasons for this, one being that traditional machine learning methods are still relatively difficult to learn contextual information about gene sequences compared with deep sequence models. Although we manually extracted many sequence preference features, they are indeed insufficient, compared to the features constructed through extensive exploration by the deep learning algorithm itself and combination of each sequence. However, there are also drawbacks to using deep learning techniques in that the features automatically extracted by the models are often uninterpretable or difficult to interpret clearly. Another reason is that the XGB4mcPred predictor uses a much smaller benchmark dataset compared to the Deep4mC and DeepTorrent predictors, which may limit the ability of the XGB4mcPred predictor to learn more information during the training process.

We also attribute the success of XGB4mcPred to two factors, one of which is the clear preference of nucleotide sequences between the non-4mC sites and 4mC sites in these six datasets from multiple species according to the previous analysis. This study used One-Hot encoding to encode original DNA sequences directly, and also encode adjacent information and spaced information of different spaced sizes. Thus, we retained the most primitive and simplest sequence information. Subsequently, we used a two-step feature selection method to remove the redundant features after encoding, making the DNA sequence information representation of these features further improved. The other factor is that we used the XGBoost algorithm to learn these DNA site sequences. The core idea of the XGBoost algorithm is to obtain a new classifier by focusing the training on the DNA sequence information of the 4mC sites misclassified by the existing base classifier, thus obtaining better results compared to other classifiers. Therefore, after the comparative analysis of the above components, we can conclude that our proposed XGB4mcPred predictor will become an essential computational tool for identifying 4mC sites.

3.7. Cross-Species Testing between Six Species

Finally, we also conducted cross-species tests on six predictors to verify the generalization ability of the predictors again. Figure 10 shows the heatmap of the accuracy performance of six independent test sets on the six predictors, where the shade of the color indicates the superiority of the accuracy of the prediction. Therefore, it can be found that, for the predictors of A. thaliana and D. melanogaster, the cross-species generalization ability of both would be relatively better. In particular, the prediction generalization accuracy of the D. melanogaster species predictor on the independent datasets of A. thaliana species, E. coli species, and G. pickeringii species reached 82.2%, 79.7%, and 78.5%, respectively. For the predictors of G. pickeringii and G. subterruneus, their cross-species generalization performance was relatively poor. Specifically, the independent datasets of C. elegans species and D. melanogaster species did not have high generalization accuracy on these two predictors, with only 63.5% and 64.0% on G. subterruneus, and 64.9% and 63.6% on G. pickeringii. More detailed results are shown in Table 5.

Table 5. The accuracy of predictors for six species in cross-species testing.

| Predictors          | E. col. Dataset | C. ele. Dataset | D. mel. Dataset | A. tha. Dataset | G. sub. Dataset | G. pic. Dataset |
|---------------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|
| E. coli             | 0.862           | 0.680           | 0.824           | 0.762           | 0.689           | 0.725           |
| C. elegans          | 0.710           | 0.900           | 0.727           | 0.670           | 0.630           | 0.643           |
| D. melanogaster     | 0.797           | 0.683           | 0.918           | 0.822           | 0.721           | 0.785           |
| A. thaliana         | 0.735           | 0.671           | 0.826           | 0.858           | 0.681           | 0.725           |
| G. subterruneus     | 0.678           | 0.635           | 0.640           | 0.679           | 0.814           | 0.720           |
| G. pickeringii      | 0.690           | 0.649           | 0.636           | 0.674           | 0.749           | 0.820           |
3.8. Challenges and Future Work

Thus, based on the discussion in the above subsections, for the method proposed in this study, our method indeed gained an effective improvement compared to machine learning methods. However, the performance of our predictor was inferior to them compared to the latest deep learning methods. Additionally, the generalization performance of the predictors for G. pickeringii species and G. subterraneus species was also inferior to the generalization performance of the predictors for the other four species. We speculate that this may be due to the weakness of our manually extracted sequence features compared to those automatically extracted by deep learning techniques, and it was found that G. pickeringii species and G. subterraneus species have nucleotide sequence preferences by analyzing nucleotide preferences. Compared with other species, the sequence preference is more complicated, resulting in a weaker generalization ability of these two models. Therefore, we believe that the model we constructed still has much room for improvement, especially in terms of feature extraction and algorithmic learning techniques. In our subsequent work, we will also think about how to use artificial features combined with deep learning techniques for modeling and try to model the DNA 4mC site identification directly across species instead of modeling each species separately to build predictors.

4. Conclusions

In this study, the proposed XGB4mcPred predictor is a novel predictor for the prediction of 4mC sites. It is a single model. This predictor encodes different types of DNA sequence information based on One-Hot encoding and illustrates the effectiveness of this coding approach by analyzing nucleotide composition preferences. The feature is optimized in two steps using the XGBoost classifier to improve the sequence information representation. By analyzing the spatial distribution of non-4mC sites and 4mC sites, we found that the non-4mC site sequences and 4mC site sequences were steadily separated, thus improving the predictive performance. Then, KNN, SVM, RF, and XGBoost classifiers were used to train and adjust the hyperparameters of the determined optimal feature subset, and the model of XGBoost training was finally determined as the optimal classifier. Two feature fusion strategies, the parallel fusion strategy and serial fusion strategy, were compared and analyzed, and finally, the serial fusion strategy was determined. Finally, using the optimal predictor XGB4mcPred in 10-fold cross-validation and independent test evaluation shows that, compared with the state-of-the-art classifiers determined by multiple feature extraction methods, multiple classifiers, or different ensemble methods, the proposed XGB4mcPred predictor gets different degrees of improvement. It is anticipated that XGB4mcPred will become an essential computational tool for identifying 4mC sites in G.pickerings, G.subterraneus, E.coli, C.elegans, A.thaliana, and D.melanogaster. XGB4mcPred
will hopefully become a useful tool for 4mC site identification in the future. The user-friendly webserver that the XGB4mcPred predictor was developed with can be accessed for free at http://www.xwanglab.com/XGB4mcPred/ (accessed on 19 September 2021). Additionally, we made our source code publicly available and stored it under our GitHub https://github.com/lyn-007/XGB4mcPred (accessed on 19 September 2021). The code includes the processing of the dataset, the extraction of features, the training of the model, and the prediction of the final model. In this way, we hope that readers will be able to make predictions not only on the website predictor we provide, but also to replicate our experiments and make predictions on their own computers.

Supplementary Materials: The following are available online at https://www.mdpi.com/article/10.3390/a14100283/s1, Figure S1: Sequence logos between 4mC sites and non-4mC sites of Caenorhabditis elegans, Arabidopsis thaliana, and Geobacter pickeringii, Figure S2: t-SNE visualization of the C. elegans, A. thaliana, and G. pickeringii datasets in a two-dimensional feature space, Table S1: Performances of the proposed XGB4mcPred predictor and four state-of-the-art predictors on six benchmark datasets from different species.

Author Contributions: Formal analysis, X.L. and R.W.; Methodology, X.W.; Project administration, X.W.; Validation, L.-J.H. and Z.-Y.D.; Writing—original draft, X.L.; Writing—review and editing, X.W., X.L. and K.-Q.F. All authors have read and agreed to the published version of the manuscript.

Funding: This work is supported in part by the National Natural Science Foundation of China (No. 61402422), the Key Science and Technology Development Program of Henan Province (No. 202102210144), the Training Program of Young Backbone Teachers in Colleges and Universities of Henan Province (No. 2019GGJS132), and the Doctorate Research Funding of Zhengzhou University of Light Industry (No. 2013BSJ082).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Acknowledgments: The authors would like to acknowledge National Natural Science Foundation of China, Department of Science and Technology of Henan Province, Department of Education of Henan Province, and Zhengzhou University of Light Industry for their financial support.

Conflicts of Interest: The authors declare no conflict of interest.

References
1. Rathi, P.; Maurer, S.; Summerer, D. Selective recognition of N4-methylcytosine in DNA by engineered transcription-activator-like effectors. Philos. Trans. R. Soc. Lond. 2018, 373, 1748. [CrossRef]
2. Blow, M.J.; Clark, T.; Daum, C.G.; Deutschbauer, A.M.; Fomenkov, A.; Fries, R.; Froula, J.; Kang, D.D.; Malmstrom, R.; Morgan, R.D.; et al. The Epigenomic Landscape of Prokaryotes. PLoS Genet. 2016, 12, e1005854. [CrossRef]
3. Fu, Y.; Luo, G.-Z.; Chen, K.; Deng, X.; Yu, M.; Han, D.; Hao, Z.; Liu, J.; Lu, X.; Dore, L.; et al. N6-methyldeoxyadenosine marks active transcription start sites in chlamydomonas. Cell 2015, 161, 879–892. [CrossRef] [PubMed]
4. Greer, E.L.; Blanco, M.A.; Gu, L.; Sendinc, E.; Liu, J.Z.; Ariztiázabal-Corrales, D.; Hsu, C.H.; Aravind, L.; He, C.; Shi, Y. DNA methylation on N6-adenine in C. elegans. Cell 2015, 161, 868–878. [CrossRef] [PubMed]
5. Heyn, H.; Esteller, M. An Adenine Code for DNA: A Second Life for N6-Methyladenine. Cell 2015, 161, 710–713. [CrossRef]
6. Jones, P.A. Functions of DNA methylation: Islands, start sites, gene bodies and beyond. Nat. Rev. Genet. 2012, 13, 484–492. [CrossRef] [PubMed]
7. Krolach, J.; Turner, S.W. Going beyond five bases in DNA sequencing. Curr. Opin. Struct. Biol. 2012, 22, 251–261. [CrossRef] [PubMed]
8. Davis, B.M.; Chao, M.C.; Waldor, M.K. Entering the era of bacterial epigenomics with single molecule real time DNA sequencing. Curr. Opin. Microbiol. 2013, 16, 192–198. [CrossRef] [PubMed]
9. Ehrlich, M.; Wilson, G.G.; Kuo, K.C.; Gehrke, C.W. N4-methylcytosine as a minor base in bacterial DNA. J. Bacteriol. 1987, 169, 939–943. [CrossRef]
10. Booth, M.; Branco, M.; Ficz, G.; Oxley, D.; Krueger, F.; Reik, W.; Balasubramanian, S. Quantitative Sequencing of 5-Methylcytosine and 5-Hydroxymethylcytosine at Single-Base Resolution. Science 2012, 336, 934–937. [CrossRef]
11. Xiao, C.L.; Zhu, S.; He, M.; Chen, D.; Zhang, Q.; Chen, Y.; Yu, G.; Liu, J.; Xie, S-Q.; Luo, F.; et al. N6-Methyladenine DNA Modification in the Human Genome. Mol. Cell 2018, 71, 306–318.e7. [CrossRef] [PubMed]
12. Ko, M.; Huang, Y.; Jankowska, A.M.; Pape, U.J.; Tahliani, M.; Bandukwala, H.S.; An, J.; Lamperti, E.D.; Koh, K.P.; Ganetzky, R.; et al. Impaired hydroxylation of 5-methylcytosine in myeloid cancers with mutant TET2. Nature 2010, 468, 839–843. [CrossRef]
13. Schweizer, H.P. Bacterial genetics: Past achievements, present state of the field, and future challenges. Biotechnology 2008, 44, 633–641. [CrossRef] [PubMed]
14. Cheng, X. DNA modification by methyltransferases. Curr. Opin. Struct. Biol. 1995, 5, 4–10. [CrossRef]
15. Flusberg, B.A.; Webster, D.R.; Lee, J.H.; Travers, K.J.; Olivares, E.C.; Clark, T.; Korlach, J.; Turner, S.W. Direct detection of DNA methylation during single-molecule, real-time sequencing. Nat. Methods 2010, 7, 461–465. [CrossRef]
16. Yu, M.; Ji, L.; Neumann, D.A.; Dae-Hwan, C.; Joseph, G.; Janet, W.; He, C.; Schmitz, R.J. Base-resolution detection of N4-methylcytosine in genomic DNA using 4mC-Tet-assisted-bisulfite-sequencing. Nucleic Acids Res. 2015, 21, e148.
17. Chen, W.C.; Yang, H.; Feng, P.M.; Ding, H.; Lin, H. idNA[4mC]: Identifying DNA N4-methylcytosine sites based on nucleotide chemical properties. Bioinformatics 2017, 33, 3518–3523. [CrossRef]
18. 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. Nucleic Acids Res. 2017, 45, D85–D89. [CrossRef] [PubMed]
19. He, W.; Jia, C.; Zou, Q. 4mCPred: Machine learning methods for DNA N4-methylcytosine sites prediction. Bioinformatics 2019, 35, 593–601. [CrossRef] [PubMed]
20. Wei, L.; Luan, S.; Nagai, L.A.E.; Su, R.; Zou, Q. Exploring sequence-based features for the improved prediction of DNA N4-methylcytosine sites in multiple species. Bioinformatics 2018, 35, 1326–1333. [CrossRef]
21. Manavalan, B.; Basith, S.; Shin, T.H.; Wei, L.; Lee, G. Meta-4mCpred: A sequence-based meta-predictor for accurate DNA 4mC site prediction using effective feature representation. Mol. Ther. Nucleic Acids 2019, 16, 733–744. [CrossRef] [PubMed]
22. 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, 4930–4937. [CrossRef] [PubMed]
23. Khanal, J.; Nazari, I.; Tayara, H.; Chong, K.T. 4mCCNN: Identification of N4-Methylcytosine sites in prokaryotes using convolutional neural network. IEEE Access 2019, 7, 145455–145461. [CrossRef]
24. Xu, H.; Jia, P.; Zhao, Z. Deep4mC: Systematic assessment and computational prediction for DNA N4-methylcytosine sites by deep learning. Brief. Bioinform. 2021, 22, 099. [CrossRef] [PubMed]
25. 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. Brief. Bioinform. 2021, 22, 124. [CrossRef] [PubMed]
26. Chen, T.; Guestrin, C. XGBoost: A Scalable Tree Boosting System. In Proceedings of the 22nd ACM SIGKDD International Conference, San Francisco, CA, USA, 13–17 August 2016; ACM: New York, NY, USA, 2016; pp. 785–794.
27. Vacic, V.; Iakoucheva, L.; Radijovic, P. Two Sample Logo: A graphical representation of the differences between two sets of sequence alignments. Bioinformatics 2006, 22, 1536–1537. [CrossRef]
28. Maaten, L.; Hinton, G. Visualizing data using t-SNE. J. Mach. Learn. Res. 2008, 9, 2579–2605.