Tissue specific prediction of N⁶-methyladenine sites based on an ensemble of multi-input hybrid neural network

CANGZHI JIA¹; DONG JIN¹; XIN WANG¹; QI ZHAO²,*

¹ School of Science, Dalian Maritime University, Dalian, 116026, China
² School of Computer Science and Software Engineering, University of Science and Technology Liaoning, Anshan, 114051, China

Key words: M6A sites, Deep hybrid neural networks, Ensemble model, Feature selection

Abstract: N⁶-Methyladenine is a dynamic and reversible post translational modification, which plays an essential role in various biological processes. Because of the current inability to identify m6A-containing mRNAs, computational approaches have been developed to identify m6A sites in DNA sequences. Aiming to improve prediction performance, we introduced a novel ensemble computational approach based on three hybrid deep neural networks, including a convolutional neural network, a capsule network, and a bidirectional gated recurrent unit (BiGRU) with the self-attention mechanism, to identify m6A sites in four tissues of three species. Across a total of 11 datasets, we selected different feature subsets, after optimized from 4933 dimensional features, as input for the deep hybrid neural networks. In addition, to solve the deviation caused by the relatively small number of experimentally verified samples, we constructed an ensemble model through integrating five sub-classifiers based on different training datasets. When compared through 5-fold cross-validation and independent tests, our model showed its superiority to previous methods, im6A-TS-CNN and iRNA-m6A.

Introduction

There are more than 160 identified types of RNA post-transcriptional modifications. Among them, the 5' cap and 3' poly modifications play important roles in transcriptional regulation, while the function of internal modification is maintaining the stability of mRNA in eukaryotes (Cao et al., 2016; Yan et al., 2021). One of the most common internal modifications is N⁶-Methyladenine (m6A). Since discovered in the 1970s, it has been observed in a wide range of eukaryotes, including yeast, Arabidopsis thaliana, Drosophila, and mammals, as well as in the RNA of viruses (Cao et al., 2016; Yang et al., 2020). N⁶-Methyladenine is a dynamic, reversible post translational modification, and is essential in post transcriptional regulation, regulating gene expression, splicing, editing RNA and maintaining genomic stability (Cao et al., 2016). However, m6A modifications were considered static and unalterable, owing to both the ignorance of m6A demethyllating enzymes and the short lifetime of most RNA species (median mammalian RNA half-lives are approximately 5 h) (Cao et al., 2016; Yan et al., 2021). The inability to identify m6A-containing mRNAs has also hindered investigation into their biological roles.

Developing computational tools for predicting m6A sites from DNA sequences could help overcome above-mentioned problems (Zhao et al., 2019; Li et al., 2018; Wei et al., 2018; Chen et al., 2017; Xing et al., 2017; Shahid and Maqsood, 2018; Wei et al., 2016; Qi et al., 2019; Liu et al., 2016; Chen et al., 2018). Computational methods to identify m6A sites can be classified as either shallow or deep learning, according to the classification algorithm adopted.

There are several representative examples of classification models based on shallow learning. Feng et al. (2019) integrated nucleotide physicochemical properties into PseKNC (Pseudo K-tuple Nucleotide Composition) and SVM (Support vector machine) so as to build a prediction tool called iDNA6mA-PseKNC. Another prediction model, SDM6A, was developed by Basith et al. (2019) to identify m6A sites in the rice genome. Basith et al. (2019) used numerical representations of nucleotides, mono-nucleotide binary encoding, di-nucleotide binary encoding, local position-specific di-nucleotide frequency, ring-function hydrogen chemical properties and K-nearest neighbor in order to select features by F-score. Then, they tried four traditional machine learning (ML) classifiers, namely SVM, ERT (extremely randomized tree), RF (random forest) and XGB (extreme

*Address correspondence to: Qi Zhao, zhaooqi@lnu.edu.cn
Received: 12 April 2021; Accepted: 02 July 2021

Doi: 10.32604/biocell.2022.016655

www.techscience.com/journal/biocell

This work is licensed under a Creative Commons Attribution 4.0 International License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.
offering insight into distinguishing m6A modi-
able to identify m6A sites in multiple species. Succeeded in constructing the iN6-Methyl model, which was
to extract features from sequences automatically, and
also the natural language processing model Word2Vec in
various tissues of spatial speci-
Nazari et al. (2019) used not only a convolutional neural network but
and a convolutional neural network (CNN). Liu et al.
could predict m6A sites by integrating one-hot encoding
model to construct an i6mA-Fuse classi-
have also been developed. Tahir et al.
Scores of these
potential compositions) to build
neither model gave satisfactory performance because of the
feature extraction and classi-
its deep feature learning ability should be promising.
To address these limitations, we proposed a novel
computation model, considering three kinds of feature
descriptors: one-hot encoding, sequence features, and order features.
Algorithm (Zhang and Liu, 2019). In this study, we
sequences automatically, and
in the DNA sequence.
For all the 11 datasets, our model gave the best
performance with measures of accuracy and Matthews
correlation coefficient. In addition, we visualized the analysis
results of the brain in human, mouse and rat using
t-distributed Stochastic Neighbor Embedding (t-SNE). Fig. 1
demonstrates the design and optimization process of our model.

Materials and Methods

Datasets

In this study, we trained and evaluated our model on the
benchmark datasets containing a total of 11 training and 11
testing datasets from human, mouse and rat, which were
also used in iRNA-m6A and im6A-TS-CNN models (Dao et al.,
Liu et al., 2020). Each dataset contains the same
number of positive and negative samples. Each sample is a
41nt-length RNA sequence with Adenine in the center.
Detailed information about these datasets can be found in
the work of Dao et al. (2020).

Feature extraction and feature selection

It is vital to extract efficacious features when developing new
computational model based on machine or deep learning
algorithms (Zhang and Liu, 2019). In this study, we
extracted three categories of features from the sequence:
one-hot encoding, sequence features, and order features.

One-hot encoding

Given a DNA sequence D, its intuitive expression is

\[ D = R_1R_2R_3R_4R_5R_7 \ldots R_9 \]  

where \( R_i \) represents the \( i \)-th nucleic acid residue at position \( i \)
in the DNA sequence.

Each sequence with 41 nt is represented with a 41 \times 41
vector, in which (1, 0, 0, 0) stands for G, (0, 1, 0, 0) stands for
C, (0, 0, 1, 0) stands for U, and (0, 0, 0, 1) stands for A.

Sequence features

Transforming a DNA sequence sample into a vector based on its
sequence characteristic composition is a simple but
universal strategy, which can capture significant biological
information (Zhen et al., 2020; Zou et al., 2019). iLearn is a
comprehensive and versatile Python-based toolkit including a
variety of descriptors for DNA, RNA and proteins (Zhen et al.,
2020). We used iLearn to calculate and extract four
types of features: nucleic acid composition, binary electron-
ion interaction pseudopotentials, autocorrelation and cross-
v covariances, pseudonucleic acid composition, and achieved a
total of 1325 features. The names and dimensions of features
used in this section are listed in Table 1. As for the
specific definitions of these features, please refer to (Zhen
et al., 2020; Zou et al., 2019).

K-tuple nucleotide frequency pattern

KNFP integrates the information from K-mer as well as one-
hot encoding, and can compensate for insufficient short-range
or local sequence order information effectively (Yang et al.,
2020). It has been used to identify protein-RNA binding
sites and protein-circRNA interaction sites (Yang et al.,
2020). K-mer can map any DNA sequence to a vector with
4^k dimensions as follows:
FIGURE 1. The framework of our model.

TABLE 1
The information of features set in this study

| Descriptor groups                  | Descriptor                                                                 | Dimension |
|-----------------------------------|---------------------------------------------------------------------------|-----------|
| Nucleic acid composition          | Nucleic Acid Composition (NAC)                                           | 4         |
|                                  | Enhanced Nucleic Acid Composition (ENAC)                                  | 148       |
|                                  | Di-Nucleotide Composition (DNC)                                           | 16        |
|                                  | Tri-Nucleotide Composition (TNC)                                          | 64        |
|                                  | Composition of k-spaced Nucleic Acid Pairs (CKSNAP)                        | 64        |
|                                  | Basic kmer (Kmer)                                                         | 84        |
|                                  | Reverse Compliment Kmer (RCKmer)                                         | 12        |
| Binary                            | Binary (Binary)                                                          | 164       |
| Electron-ion interaction pseudopotentials | Electron-ion interaction pseudopotentials of trinucleotide (EIIP),       | 41        |
|                                  | Electron-ion interaction pseudopotentials of trinucleotide (PseEIIP)      | 64        |
| Autocorrelation and cross-covariance | Dinucleotide-based Auto Covariance (DAC)                                 | 30        |
|                                  | Dinucleotide-based Cross Covariance (DCC)                                 | 150       |
|                                  | Dinucleotide-based Auto-Cross Covariance (DACC)                          | 180       |
|                                  | Trinucleotide-based Auto Covariance (TAC)                                 | 10        |
|                                  | Trinucleotide-based Cross Covariance (TCC)                                | 10        |
|                                  | Trinucleotide-based Auto-Cross Covariance (TACC)                         | 20        |
| Pseudo nucleic acid composition   | Pseudo Dinucleotide Composition (PseDNC)                                 | 18        |
|                                  | Pseudo k-tuple Composition (PseKNC)                                      | 66        |
|                                  | Parallel Correlation Pseudo Dinucleotide Composition (PCPseDNC)           | 18        |
|                                  | Parallel Correlation Pseudo Trinucleotide Composition (PCPseTNC)          | 66        |
|                                  | Series Correlation Pseudo Dinucleotide Composition (SCPseDNC)             | 28        |
$R = [\varphi_1 \varphi_2 \cdots \varphi_u \cdots \varphi_k]^T$ \hfill (2)

where $\varphi_u (u = 1, 2, \cdots, 4^k)$ is the frequency of the $u$-th k-mer along the sequence ($k = 1, 2, 3$ in this work).

On one hand, $R$ can be transformed to a diagonal matrix $R_D$, if multiplied by the identity matrix. On the other hand, for a DNA sequence $D$ of length $L$, the number of k-mer is $L - k + 1$. Each k-mer can be encoded as a one-hot vector with dimension of $4^k$. The product of $(L - k + 1) \times 4^k$ matrix ($M$) and $R_D$ is KNFP.

For example, given a sequence $S = 'ACGACGA'$, 1-mer is encoded as one-hot vectors: $G = (1, 0, 0, 0)$, $C = (0, 1, 0, 0)$, $U = (0, 0, 1, 0)$, and $A = (0, 0, 0, 1)$. Then according to the position information, $S$ can be transformed to a matrix as follows:

$$
M = \begin{pmatrix}
0 & 0 & 0 & 1 \\
0 & 1 & 0 & 0 \\
1 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 \\
0 & 1 & 0 & 0 \\
1 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 \\
0 & 0 & 0 & 0 \\
\end{pmatrix}
$$

For $S$, the frequency vector of 1-mer ($G, C, U, A$) is $R = (0.25, 0.25, 0.25, 0.5)$. Then through multiplying $R$ by an identity matrix, it is converted to a diagonal matrix as follows:

$$
R_D = \begin{pmatrix}
0.25 & 0 & 0 & 0 \\
0 & 0.25 & 0 & 0 \\
0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0.5 \\
\end{pmatrix}
$$

The KNFP $= M \times R_D$ is given as

$$
\begin{pmatrix}
0 & 0 & 0 & 1 \\
0 & 1 & 0 & 0 \\
1 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 \\
0 & 1 & 0 & 0 \\
1 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 \\
0 & 0 & 0 & 1 \\
\end{pmatrix} \times
\begin{pmatrix}
0.25 & 0 & 0 & 0 \\
0 & 0.25 & 0 & 0 \\
0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0.5 \\
\end{pmatrix} = \begin{pmatrix}
0 & 0 & 0 & 0.5 \\
0 & 0.25 & 0 & 0 \\
0.25 & 0 & 0 & 0 \\
0 & 0.25 & 0 & 0 \\
0.25 & 0 & 0 & 0.5 \\
0 & 0 & 0 & 0.5 \\
\end{pmatrix}
$$

**Feature optimization**

Experiments have shown that excessive feature information can interfere with the performance of classifiers. Therefore, feature selection methods should be applied to find the most informative features for training classifiers and thus reduce the dimensionality of the feature vector. F-score has been extensively applied in bioinformatics because of its effectiveness in balancing accuracy and stability (Bui et al., 2016; Li et al., 2018).

The F-score of the $j$-th feature is defined as

$$
F - \text{score}(j) = \frac{(\bar{x}_j^{(+)} - \bar{x}_j)^2 + (\bar{x}_j^{(-)} - \bar{x}_j)^2}{\frac{1}{m-1}\sum_{i=1}^{m}(x_{i,j}^{(+)} - \bar{x}_j^{(+)})(x_{i,j}^{(-)} - \bar{x}_j^{(-)})^2}
$$

where $\bar{x}_j$, $\bar{x}_j^{(+)}$ and $\bar{x}_j^{(-)}$ denote average values of the $j$-th feature in the combined positive and negative training datasets, the positive datasets, and the negative datasets, respectively. $m^+$ is the total number of positive samples; $m^-$ is the total number of negative samples; $\bar{x}_j$ represents the $j$-th feature of the $k$-th positive sample and $\bar{x}_j^{(+)}$ represents the $j$-th feature of the $k$-th negative sample. The higher the F-score is, the more useful the corresponding feature is for the classification.

**Multi-input hybrid neural network**

The model architecture consisted of a multi-channel CNN, a capsule network and a BiGRU network. Each of these three networks has, respectively, been shown to be effective in object detection (Li et al., 2018), protein post-translational modification site prediction (Wang et al., 2019) and social bots detection (Wu et al., 2020). The structure of the multi-input hybrid neural network is shown in Fig. 2.

**Multi-channel CNN**

The input of the multi-input hybrid neural network is the one-hot encoding, sequence features and KNFP, respectively. For each type of features, we applied 32 convolution filters and performed batch normalization to readjust the data distribution. The input of a batch in the neural network is $X = [x_1, x_2, \cdots, x_n]$, where $x_i$ represents a sample and $n$ is the batch size.

The mean value of elements in each batch is obtained by:

$$
\mu_B = \frac{1}{n} \sum_{i=1}^{n} x_i
$$

Then, the variance of a batch is calculated by

$$
\sigma_B^2 = \frac{1}{n} \sum_{i=1}^{n} (x_i - \mu_B)^2
$$

This allows us to normalize each element:

$$
\tilde{x}_i = \frac{x_i - \mu_B}{\sqrt{\sigma_B^2 + \varepsilon}}
$$

The final output of the network is given by

$$
y_i = \gamma_i \cdot \tilde{x}_i + \beta_i
$$

where $\varepsilon$ is a small positive number used to prevent the denominator from being 0.

Finally, we merge three outputs using the Swish activation function $\sigma$, which is defined as follows:

$$
\sigma(x) = \frac{1}{1 + \exp(-x)}
$$

Because of the limited size of our datasets, we add a $1 \times 1$ multi-channel CNN to enhance representational capabilities.
of the model. The 1 × 1 convolution is used to maintain the size of the feature map and integrate the information by linearly weighting the input feature map of each channel. With additional layers of such 1 × 1 convolution, the final extracted features would become more concise.

Capsule network
The capsule network (CapsNet) was proposed by Sabour et al. (2017) and applied in stance detection (Zhao and Yang, 2020), image recognition (Qian et al., 2020) and automated classification (Mobiny et al., 2019). Since the capsule network collects location information, it can learn a good representation from a small amount of data. We use the capsule network and focus on the hierarchical relationship of local features. The output of the multi-channel CNN is adopted as the input vector of the capsule network. We make an affine transformation of the input vector as follows:

$$\tilde{u}_{ij} = W_{ij}u_i$$

where $W_{ij}$ is the weight matrix that needs to be trained and $u_i$ is the input vector of the capsule neural network.

Next, the weighted sum is applied to all the prediction vectors as follows:

$$s_j = \sum_i c_{ij}\tilde{u}_{ij},$$

where $c_{ij}$ is the coupling coefficient in the dynamic routing process.

Finally, we obtain output vectors through a non-linear activation function as follows:

$$v_j = \frac{||s_j||^2}{1 + ||s_j||^2} s_j.$$  

BiGRU network
The third segment of this model is the BiGRU network, which helps to extract deep-level features of sequences. The current hidden layer state of the BiGRU is determined by three factors: the current input $x_t$, the output of the forward hidden layer state at time-step $(t-1)$ $h_{t-1}^{\rightarrow}$ and the output of the reverse hidden layer state $h_{t-1}^{\leftarrow}$. BiGRU can be regarded as two GRUs, so the state of hidden layer $h_t$ at time-step $t$ can be obtained by the weighted sum of the forward hidden layer state $h_{t-1}^{\rightarrow}$ and reverse hidden layer state $h_{t-1}^{\leftarrow}$:

$$h_t = \text{GRU}(x_t, h_{t-1}^{\rightarrow})$$

$$h_t = \text{GRU}(x_t, h_{t-1}^{\leftarrow})$$

$$h_t = w_t \cdot h_t + v_t \cdot h_t + b_t,$$

where GRU() represents a nonlinear transformation of the input word vector; $w_t$ and $v_t$ are the weighted matrices; and $b_t$ is the bias term.

Parameter setting
Considering the number of datasets and the precision of the model, three feature maps were obtained from batch normalization and 1D convolution with 32 filters (kernel size = 3). The multi-channel CNN contained three 1 × 1 convolution layers and took the Swish as activation function. Considering time cost, we only employed one capsule layer with 14 num_capsule (dim_capsule = 41, routings = 3). The BiGRU had 32 hidden units followed by a fully connected layer and used the ReLU activation function. We also used dropout with a keep probability of 0.3 to prevent the model from over fitting. For stochastic gradient descent, we selected the Adam optimization algorithm. The entire program was written in Python 3.6.

Performance assessment
To evaluate the performance of our prediction model, we used four measurements including accuracy (Acc), sensitivity (Sn), specificity (Sp), and Matthew’s correlation coefficient (MCC) on 5-fold cross-validation and independent dataset tests. The formulas are provided as follows:

$$Sp = \frac{TN}{TP+FP}$$
$$Sn = \frac{TP}{FP+TP}$$
$$Acc = \frac{TP+TN}{TP+TN+FP+FN}$$
$$MCC = \frac{(TP+FN)(FP+TN)(TP+FP)(FN+TN)}{\sqrt{(TP+FN)(FP+TN)(TP+FP)(FN+TN)}}$$

where TP, TN, FP, and FN represent the number of true positives, true negatives, false positives, and false negatives, respectively.

Results
We observed a deviation in the process of 5-fold cross-validation test (Table S1), probably due to the limited number of experimentally verified samples. A general strategy to solve the problem of insufficient samples is to construct an ensemble prediction model. On this purpose,
we randomly divided the positive and negative training datasets into five mutually exclusive parts of similar size. And then, we selected the combination of four parts as a new training dataset, while the remaining one part was adopted as validation test dataset to train and optimize the three-layer hybrid neural network at each time. Thus, we got five sub models which were then integrated into a novel ensemble prediction model based on a majority voting strategy.

In order to verify the effectiveness of the ensemble model, we compared it with two previous prediction methods: im6A-TS-CNN and iRNA-m6A. The same training and independent datasets were used for our model, im6A-TS-CNN and iRNA-m6A; therefore, both 5-fold cross-validation and independent test could be used to evaluate these methods objectively. There was a total of 132 results from 11 datasets involving four indicators (Sn, Sp, Acc, MCC), among which our model achieved superior predictive performance as measured by average MCC and Acc. Specifically, on the 5-fold cross validation, for Homo sapiens, our model gave MCC = 0.581, vs. 0.550 for the second-placed im6A-TS-CNN; for Mus musculus, our model gave MCC = 0.558 vs. 0.517 for the second-placed im6A-TS-CNN; for Rattus norvegicus, our model reached MCC = 0.626 vs. 0.600 for the second-placed im6A-TS-CNN. On the independent dataset, for Homo sapiens, our model showed MCC = 0.572 vs. 0.547 for the second-placed im6A-TS-CNN; for Mus musculus, our model gave MCC = 0.546 vs. 0.525 for the second-placed im6A-TS-CNN; for Rattus norvegicus, our model reached MCC = 0.617 vs. 0.604 for the second-placed im6A-TS-CNN.

To observe the comparison results intuitively, we showed MCC values of these three models in Figs. 3 and 4. Moreover, the comparison results measured with other indicators are provided in Tables S2 and S3.

**Discussion**

**Effectiveness of feature selection**

If too many features are extracted, the generalizability of the model will be weakened. Thus it is important to determine the appropriate step size for feature selection. As the dimension of the input matrix was $41 \times N$, we chose the step size as 41 and evaluated the performance of our model with feature matrices of different dimensions ($41 \times N$, $N \in \{5, 6, 7, \ldots, 24, 25\}$) on 5-fold cross-validation tests successively. To reduce the deviation caused by the fluctuation of the neural network, we ran each test three times for the 11 datasets from the brain, liver, kidney, heart, and tests of Homo sapiens, Mus musculus, and Rattus norvegicus. The optimal feature subset was finalized according to the average accuracy. The detailed feature selection results are shown in Fig. 5. As should be noticed, dimensions of optimal features were various for different datasets; their corresponding dimensions are listed in Table 2. These results indicate that it is necessary to establish a specialized model for each tissue type in each species.

**Parameter selection**

Generally, there are two ways to select parameters, i.e., empirical choice and Bayesian optimization. With Homo sapiens as an example, we tried both methods to find the most suitable parameters. The initial parameters were set based on a previous study to compare the prediction results roughly. Then, if the prediction model was under fitting, we attempted to add more convolution kernels and neurons; otherwise if the prediction model was over fitting, we attempted to reduce the number of convolution kernels and neurons. In addition, batch normalization, dropout, and regularization were introduced to avoid over fitting during the optimizing process. Alternatively, Bayesian optimization (Snoek et al., 2012) was used to tune the key parameters including batch size, dropout rate, filter1, filter2, pool_size and etc. Finally, the optimal parameters were selected according to the AUC value. The Baysian optimization of parameters in models for Homo sapiens and corresponding AUC values are provided in Table S4 and Fig. 6A. We compared the best results obtained by the two methods and concluded the result given by empirical adjustment parameters showed more advantages (Fig. 6B). Thus, for Mus musculus and Rattus norvegicus, we used the empirical method to determine the parameters in neural network.

**Comparison of different classifiers**

To verify the effectiveness of hybrid neural network, we compared its prediction performance with several traditional classification algorithms including logistic regression, decision tree, support vector machine (SVM), random forest (RF), gradient boosting decision tree (GBDT), extreme gradient boosting (XGBoost) and light gradient boosting machine (LightGBM) for Homo sapiens. The average accuracies of 5-fold cross-validation tests obtained by the

---

**FIGURE 3.** Comparison with MCC measure of different models on 5-fold cross validation test.
seven algorithms are listed in Table 3. On three datasets of *Homo sapiens*, our model achieves the mean accuracy of 74.29%, 1.28% higher than the second best algorithm, LightGBM, which demonstrates that hybrid neural network is capable of improving the recognition performance for m6A sites of various tissues in different species.

**FIGURE 4.** Comparison with MCC measure of different models on independent tests.

**FIGURE 5.** The accuracy comparison of different feature dimension on 11 datasets of three species.
Visualization of feature representations
To observe the effectiveness of extracted features intuitively, we applied t-distributed stochastic neighbor embedding (t-SNE) to visualize the feature representations. We took the brain tissue as an example and demonstrated the features after mapped through the concatenate layer and the attention layer. Each dot in Fig. 6C represents a sample, with purple dots representing m6A sites and yellow dots representing non-m6A sites. The overlapping of the two sample types on the left side of Fig. 6C indicates that it is difficult to distinguish m6A sites from non-m6A sites. However, the features were relatively separated, as shown on the right side of Fig. 6C, after selected and processed by the deep hierarchical network, which was, multi-channel CNN, capsule network and BiGRU network with the self-attention mechanism. The t-SNE plots indicated that the hybrid deep hierarchical networks could learn sequence motif information from selected features. But there are still some overlaps between the two types, indicating that our model is not completely specific for all m6A sites. This fact is consistent with the need to establish specialized models for each tissue type.

### TABLE 2

| Datasets | Feature Dimension | Acc  |
|----------|-------------------|------|
| h_b      | 41 × 19           | 73.91% |
| h_k      | 41 × 5            | 80.76% |
| h_l      | 41 × 5            | 81.49% |
| m_b      | 41 × 5            | 79.86% |
| m_h      | 41 × 12           | 75.87% |
| m_k      | 41 × 12           | 81.87% |
| m_l      | 41 × 11           | 73.43% |
| m_t      | 41 × 15           | 77.23% |
| r_b      | 41 × 15           | 78.23% |
| r_k      | 41 × 13           | 83.25% |
| r_l      | 41 × 7            | 82.43% |

### TABLE 3

Comparison of different classifiers for identifying m6A sites on 5-fold cross-validation

| Method                | h_b  | h_k  | h_l  |
|-----------------------|------|------|------|
| Support Vector Machines | 71.82 | 79.03 | 79.90 |
| Decision Tree         | 62.75 | 71.29 | 71.24 |
| Logistic Regression   | 65.81 | 71.33 | 71.91 |
| Random forest         | 70.92 | 78.49 | 79.65 |
| GBDT                  | 73.55 | 80.31 | 80.77 |
| XGBoost               | 72.13 | 79.14 | 80.16 |
| LightGBM              | 73.01 | 80.00 | 80.68 |
| **Our model**         | **74.29** | **80.75** | **81.80** |

FIGURE 6. (continued)
Conclusion

In this work, we introduced a novel ensemble computational approach to identify m6A sites, based on three hybrid neural networks. For different tissues of different species, we selected different optimized feature subsets from 4933 features as multi-input for the deep hybrid neural networks. Our predication model consisted of a multi-channel CNN, a capsule network, and a BiGRU network with the self-attention mechanism, and was evaluated on 11 datasets. To solve the deviation caused by the relatively small number of experimentally verified samples, we constructed an ensemble model through integrating five sub-classifiers based on different training datasets. To estimate the performance of this model, comparisons were made on 11 datasets by 5-fold cross validation and independent test datasets. Results of all tests revealed that when measured with Acc and MCC, our model is superior to two previous tools, iRNA-m6A and im6A-TS-CNN. However, the specificity of our model is not satisfactory on h_b, h_k, m_h, m_k and r_b datasets. In future work, we will extract more types of information and further optimize these models.
Availability of Data and Materials: The data sets and source code used in this study are freely available at https://github.com/Dong7777/im6mA.

Author Contribution: QZ and XW conceived the project, developed the prediction method, designed and implemented the experiments, analyzed the results, and wrote the paper. DJ and CZJ implemented the experiments, analyzed the results, and wrote the paper. All authors read and approved the final manuscript.

Funding Statement: This work was supported by the National Natural Science Foundation of China (Nos. 62071079 and 61803065).

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

Basith S, Manavalan B, Shin TH, Lee G (2019). SDM6A: A web-based integrative machine-learning framework for predicting 6mA sites in the rice genome. *Molecular Therapy Nucleic Acids* 18: 131–141. DOI 10.1016/j.omtn.2019.08.011.

Bui VM, Weng SL, Lu CT, Chang TH, Weng JT, Lee TY (2016). SOHSite: Incorporating evolutionary information and physicochemical properties to identify protein S-sulfenylation sites. *BMC Genomics* 17: 1–9. DOI 10.1186/s12864-015-2299-1.

Cao G, Li HB, Yin Z, Flavell RA (2016). Recent advances in dynamic m6A RNA modification. *Open Biology* 6: 160003. DOI 10.1098/rsob.160003.

Chen W, Peng F, Yang H, Ding H, Lin H, Chou KC (2018). iRNA-3typeA: Identifying three types of modification at RNA’s adenosine sites, molecular therapy. *Molecular Therapy Nucleic Acids* 11: 468–474. DOI 10.1016/j.omtn.2018.03.012.

Chen W, Xing P, Zou Q (2017). Detecting N6-methyladenosine sites from RNA transcripts using ensemble support vector machines. *Scientific Reports* 7: 40242. DOI 10.1038/srep40242.

Duo FY, LV H, Yang YH, Zulfiqar H, Lin H (2020). Computational identification of N6-methyladenosine sites in multiple tissues of mammals. *Computational and Structural Biotechnology Journal* 18: 1084–1091. DOI 10.1016/j.csbj.2020.04.015.

Feng P, Yang H, Ding H, Lin H, Chen W et al. (2019). iDNA6mA-PseKNC: Identifying DNA N6-methyladenosine sites by incorporating nucleotide physicochemical properties into PseKNC. *Genomics* 111: 96–102. DOI 10.1016/j.ygeno.2018.01.005.

Hasan MM, Manavalan B, Shoombuatong W, Khatun MS, Kurata H (2020). i6mA-Fuse: Improved and robust prediction of DNA 6mA sites in the Rosaceae genome by fusing multiple feature representation. *Plant Molecular Biology* 103: 225–234. DOI 10.1007/s11103-020-00988-y.

Li QX, Rong CH, Cai YX, Ran S, Yi WL (2018). M6AMRFS: Robust prediction of N6-methyladenosine sites using sequence-based features in multiple species. *Frontiers in Genetics* 9: 495. DOI 10.3389/fgene.2018.00495.

Liu K, Cao L, Du P, Chen W (2020). i6mA-TS-CNN: Identifying the N(6)-methyladenine site in multiple tissues by using the convolutional neural network. *Molecular Therapy Nucleic Acids* 21: 1044–1049. DOI 10.1016/j.omtn.2020.07.034.

Liu S, Liu Z (2017). Multi-channel CNN-based object detection for enhanced situation awareness. *Sensors and Electronics Technology (SET) Panel Symposium SET-241 on 9th NATO Military Sensing Symposium.*

Liu Z, Xiao X, Yu DJ, Jia J, Qiu WR, Chou KC (2016). pRNAm-PC: Predicting N(6)-methyladenosine sites in RNA sequences via physical-chemical properties. *Analytical Biochemistry* 497: 60–67. DOI 10.1016/j.ab.2015.12.017.

Nazari I, Tahir M, Tayara H, Chong KT (2019). iN6-methyl (5-step): Identifying RNA N6-methyladenosine sites using deep learning model via cho’s 5-step rules and cho’s general PseKNC. *Chemometrics and Intelligent Laboratory Systems* 193: 103811. DOI 10.1016/j.chemolab.2019.103811.

Qi CK, Zhen W, Qing Z, Yu WX, Rong R, Liang LZ, Long SJ, João MP, Daniel JR, Jia M (2019). WHISTLE: A high-accuracy map of the human N6-methyladenosine (m6A) epitranscriptome predicted using a machine learning approach. *Nucleic Acids Research* 47: e41. DOI 10.1093/nar/gkz074.

Qian K, Tian L, Liu Y, Wen X, Bao J (2020). Image robust recognition based on feature-entropy-oriented differential fusion capsule network. *Applied Intelligence* 51: 1108–1117. DOI 10.1007/s10489-020-01873-3.

Sabour S, Frosst N, Hinton GE (2017). Dynamic routing between capsules. 31st Conference on Neural Information Processing Systems (NIPS 2017). Long Beach, CA, USA.

Shahid A, Maqpool H (2018). iMethyl-STTNC: Identification of N6-methyladenosine sites by extending the idea of SAAC into Chou’s PseAAC to formulate RNA sequences. *Journal of Theoretical Biology* 455: 205–211. DOI 10.1016/j.jtbi.2018.07.018.

Snoek J, Larochelle H, Adams RP (2012). Practical Bayesian optimization of machine learning algorithms. In: *Advances in Neural Information Processing Systems*, vol. 4.

Tahir M, Tayara H, Chong KT (2019). iDNA6mA (5-step rule): Identification of DNA N6-methyladenine sites in the rice genome by intelligent computational model via cho’s 5-step rule. *Chemometrics & Intelligent Laboratory Systems* 189: 96–101. DOI 10.1016/j.chemolab.2019.04.007.

Wang D, Liang Y, Xu D (2019). Capsule network for protein post-translational modification site prediction. *Bioinformatics* 35: 2386–2394. DOI 10.1093/bioinformatics/bty977.

Wei L, Chen H, Ran S (2018). M6APred-EL: A sequence-based predictor for identifying N6-methyladenosine sites using ensemble learning. *Molecular Therapy Nucleic Acids* 12: 635–644. DOI 10.1016/j.omtn.2018.07.004.

Wei C, Hua T, Hao L (2016). MethyRNA: A web-server for identification of N(6)-methyladenosine sites. *Journal of Biomolecular Structure & Dynamics* 35: 1–11. DOI 10.1080/07391102.2016.1157761.

Wu Y, Fang Y, Shang S, Jin J, Wang H (2020). A novel framework for detecting social bots with deep neural networks and active learning. *Knowledge-Based Systems* 211: 106525. DOI 10.1016/j.knosys.2020.106525.

Xing P, Su R, Guo F, Wei L (2017). Identifying N6-methyladenosine sites using multi-interval nucleotide pair position specificity and support vector machine. *Scientific Reports* 7: 46757. DOI 10.1038/srep46757.

Yan AC, Liang Y, Quan Z (2021). Prediction of bio-sequence modifications and the associations with diseases. *Briefings in Functional Genomics* 20: 1–18. DOI 10.1093/bfgp/elaa023.
Zhang J, Liu B (2019). A review on the recent developments of sequence-based protein feature extraction methods. Current Bioinformatics 14: 190–199. DOI 10.2174/1574893614666181212102749.

Zhao G, Yang P (2020). Pretrained embeddings for stance detection with hierarchical capsule network on social media. ACM Transactions on Information Systems 39: 1. DOI 10.1145/3412362.

Zou Q, Xing P, Wei L, Liu B (2019). Gene2vec: Gene subsequence embedding for prediction of mammalian N6-methyladenosine sites from mRNA. RNA 25: 205–218. DOI 10.1261/rna.069112.118.

Supplementary Tables

TABLE S1

| Data  | Time | Method | Sn (%) | Sp (%) | MCC  | Acc (%) |
|-------|------|--------|--------|--------|------|---------|
| h_b   | 1    | 5-fold | 80.37  | 67.73  | 49.08| 74.00  |
|       |      | independent | 81.78 | 66.18 | 48.55 | 73.98 |
|       | 2    | 5-fold | 82.03  | 65.77  | 48.48| 73.85  |
|       |      | independent | 82.99 | 64.68 | 48.50 | 73.84 |
|       | 3    | 5-fold | 80.00  | 68.48  | 48.94| 74.62  |
|       |      | independent | 81.82 | 67.42 | 49.76 | 74.62 |
|       | 4    | 5-fold | 83.61  | 64.19  | 48.79| 73.93  |
|       |      | independent | 85.12 | 63.18 | 49.51 | 74.15 |
|       | 5    | 5-fold | 79.08  | 68.73  | 48.08| 73.91  |
|       |      | independent | 80.30 | 68.09 | 48.76 | 74.20 |
| h_k   | 1    | 5-fold | 82.69  | 78.79  | 61.56| 80.73  |
|       |      | independent | 80.89 | 79.23 | 60.12 | 80.06 |
|       | 2    | 5-fold | 82.05  | 79.00  | 61.18| 80.54  |
|       |      | independent | 80.52 | 79.27 | 59.79 | 79.89 |
|       | 3    | 5-fold | 83.59  | 77.90  | 61.62| 80.75  |
|       |      | independent | 82.33 | 77.35 | 59.75 | 79.84 |
|       | 4    | 5-fold | 84.39  | 77.07  | 61.67| 80.71  |
|       |      | independent | 83.21 | 76.80 | 60.13 | 80.00 |
|       | 5    | 5-fold | 83.25  | 77.89  | 61.25| 80.59  |
|       |      | independent | 82.05 | 77.78 | 59.88 | 79.91 |
| h_l   | 1    | 5-fold | 84.85  | 78.74  | 63.75| 81.80  |
|       |      | independent | 84.13 | 77.79 | 62.05 | 80.96 |
|       | 2    | 5-fold | 84.79  | 77.79  | 62.85| 81.36  |
|       |      | independent | 84.28 | 78.06 | 62.46 | 81.17 |
|       | 3    | 5-fold | 84.26  | 78.50  | 62.92| 81.42  |
|       |      | independent | 83.79 | 78.82 | 62.68 | 81.30 |
|       | 4    | 5-fold | 83.77  | 79.36  | 63.22| 81.61  |
|       |      | independent | 83.14 | 79.23 | 62.42 | 81.19 |
|       | 5    | 5-fold | 85.02  | 77.76  | 61.36| 81.36  |
|       |      | independent | 84.05 | 77.83 | 62.00 | 80.94 |
| m_b   | 1    | 5-fold | 83.09  | 76.51  | 59.75| 79.80  |
|       |      | independent | 83.48 | 75.90 | 59.55 | 79.69 |

(Continued)
| Data | Time | Method  | Sn (%) | Sp (%) | MCC   | Acc (%) |
|------|------|---------|--------|--------|-------|---------|
| 2    |      | 5-fold  | 82.65  | 77.06  | 59.91 | 79.91   |
|      |      | independent | 82.92  | 75.85  | 58.91 | 79.38   |
| 3    |      | 5-fold  | 83.81  | 75.58  | 59.64 | 79.69   |
|      |      | independent | 84.27  | 74.64  | 59.19 | 79.46   |
| 4    |      | 5-fold  | 82.07  | 77.52  | 59.67 | 79.79   |
|      |      | independent | 82.44  | 76.02  | 58.59 | 79.23   |
| 5    |      | 5-fold  | 82.96  | 76.32  | 59.49 | 79.68   |
|      |      | independent | 83.48  | 75.90  | 59.55 | 79.69   |
| m_h  | 1    | 5-fold  | 79.77  | 71.26  | 51.59 | 75.62   |
|      |      | independent | 80.27  | 69.73  | 50.28 | 75.00   |
|      | 2    | 5-fold  | 81.56  | 70.64  | 52.33 | 76.10   |
|      |      | independent | 80.36  | 70.32  | 50.94 | 75.34   |
|      | 3    | 5-fold  | 79.82  | 71.97  | 52.23 | 75.90   |
|      |      | independent | 79.27  | 71.36  | 50.80 | 75.32   |
|      | 4    | 5-fold  | 79.53  | 72.24  | 51.96 | 75.92   |
|      |      | independent | 78.27  | 72.64  | 50.99 | 75.45   |
|      | 5    | 5-fold  | 79.29  | 72.36  | 51.89 | 75.87   |
|      |      | independent | 78.09  | 70.82  | 49.04 | 74.45   |
| m_k  | 1    | 5-fold  | 83.41  | 79.94  | 63.40 | 81.66   |
|      |      | independent | 83.05  | 79.83  | 62.91 | 81.44   |
|      | 2    | 5-fold  | 67.97  | 78.93  | 63.61 | 81.74   |
|      |      | independent | 84.19  | 78.62  | 62.90 | 81.40   |
|      | 3    | 5-fold  | 84.26  | 79.19  | 63.56 | 81.72   |
|      |      | independent | 83.63  | 78.64  | 62.35 | 81.14   |
|      | 4    | 5-fold  | 83.57  | 79.45  | 63.11 | 81.51   |
|      |      | independent | 83.91  | 78.34  | 62.34 | 81.12   |
|      | 5    | 5-fold  | 83.97  | 78.96  | 63.05 | 81.47   |
|      |      | independent | 83.76  | 78.52  | 62.36 | 81.14   |
| m_l  | 1    | 5-fold  | 77.35  | 69.45  | 47.18 | 73.37   |
|      |      | independent | 76.07  | 69.61  | 45.78 | 72.84   |
|      | 2    | 5-fold  | 77.76  | 69.04  | 47.06 | 73.43   |
|      |      | independent | 77.16  | 68.84  | 46.16 | 73.00   |
|      | 3    | 5-fold  | 81.37  | 65.03  | 47.28 | 73.25   |
|      |      | independent | 81.15  | 64.77  | 46.55 | 72.96   |
|      | 4    | 5-fold  | 76.95  | 70.87  | 47.98 | 73.92   |
|      |      | independent | 75.34  | 70.72  | 46.12 | 73.03   |
|      | 5    | 5-fold  | 79.39  | 67.63  | 47.36 | 73.48   |
|      |      | independent | 78.30  | 67.70  | 46.26 | 73.00   |
| m_t  | 1    | 5-fold  | 83.08  | 70.78  | 54.30 | 76.93   |
|      |      | independent | 83.60  | 70.29  | 54.37 | 76.94   |
|      | 2    | 5-fold  | 79.98  | 73.97  | 54.15 | 77.01   |
|      |      | independent | 80.41  | 74.20  | 54.72 | 77.31   |
|      | 3    | 5-fold  | 82.28  | 72.01  | 54.74 | 77.20   |
|      |      | independent | 81.68  | 72.23  | 54.15 | 76.96   |
|      | 4    | 5-fold  | 84.01  | 69.92  | 54.63 | 77.02   |
|      |      | independent | 84.17  | 70.17  | 54.88 | 77.17   |
### Table S1 (continued).

| Data | Time   | Method     | Sn (%) | Sp (%) | MCC  | Acc (%) |
|------|--------|------------|--------|--------|------|----------|
|      | 5      | 5-fold     | 83.67  | 70.32  | 54.55| 77.00    |
|      |        | independent | 84.06  | 70.38  | 54.96| 77.22    |
| r_b  | 1      | 5-fold     | 81.25  | 75.22  | 56.71| 78.27    |
|      |        | independent | 79.71  | 75.29  | 55.05| 77.50    |
|      | 2      | 5-fold     | 81.31  | 74.47  | 55.94| 77.87    |
|      |        | independent | 79.80  | 74.99  | 54.85| 77.39    |
|      | 3      | 5-fold     | 82.43  | 73.29  | 56.02| 77.83    |
|      |        | independent | 81.28  | 73.37  | 54.83| 77.33    |
|      | 4      | 5-fold     | 82.67  | 73.59  | 56.55| 78.10    |
|      |        | independent | 82.18  | 72.91  | 55.32| 77.54    |
|      | 5      | 5-fold     | 80.36  | 75.99  | 56.42| 78.17    |
|      |        | independent | 79.20  | 75.71  | 54.95| 77.46    |
|      | r_k    | 1          | 5-fold | 86.19  | 79.59| 65.96    |
|      |        | independent | 86.98  | 79.08  | 66.26| 83.03    |
|      | 2      | 5-fold     | 84.91  | 80.87  | 65.83| 82.89    |
|      |        | independent | 85.58  | 80.91  | 66.56| 83.25    |
|      | 3      | 5-fold     | 85.13  | 80.47  | 65.78| 82.84    |
|      |        | independent | 86.54  | 80.42  | 67.08| 83.48    |
|      | 4      | 5-fold     | 84.78  | 80.82  | 65.84| 82.84    |
|      |        | independent | 85.81  | 80.54  | 66.44| 83.17    |
|      | 5      | 5-fold     | 83.53  | 82.47  | 66.02| 83.00    |
|      |        | independent | 84.82  | 82.08  | 66.92| 83.45    |
|      | r_l    | 1          | 5-fold | 83.57  | 80.74| 64.32    |
|      |        | independent | 84.96  | 78.49  | 63.58| 81.73    |
|      | 2      | 5-fold     | 83.69  | 80.93  | 64.72| 82.26    |
|      |        | independent | 85.24  | 77.87  | 63.28| 81.56    |
|      | 3      | 5-fold     | 83.43  | 81.64  | 65.08| 82.46    |
|      |        | independent | 84.17  | 78.83  | 63.09| 81.50    |
|      | 4      | 5-fold     | 83.58  | 81.02  | 64.69| 82.29    |
|      |        | independent | 84.62  | 79.17  | 63.89| 81.90    |
|      | 5      | 5-fold     | 83.66  | 80.83  | 64.61| 82.29    |
|      |        | independent | 84.85  | 78.60  | 63.57| 81.73    |

### Table S2

Comparison of our model with im6A-TS-CNN and iRNA-m6A on 5-fold cross-validation test

| Species | Methods     | Sn (%) | Sp (%) | Acc (%) | MCC   |
|---------|-------------|--------|--------|---------|-------|
| h_b     | iRNA-m6A    | 74.79  | 66.19  | 71.26   | 0.41  |
|         | im6A-TS-CNN | 75.35  | 69.71  | 72.53   | 0.4523|
|         | our model   | 80.00  | 68.48  | 74.29   | 0.4894|
| h_k     | iRNA-m6A    | 80.85  | 76.34  | 78.99   | 0.57  |
|         | im6A-TS-CNN | 81.70  | 78.25  | 79.98   | 0.6006|
|         | our model   | 83.59  | 77.90  | 80.75   | 0.6162|
| h_l     | iRNA-m6A    | 81.32  | 78.13  | 80.13   | 0.59  |
|         | im6A-TS-CNN | 80.18  | 79.69  | 79.94   | 0.5992|
|         | our model   | 84.85  | 78.74  | 81.80   | 0.6375|
| Species | Methods          | Sn (%) | Sp (%) | Acc (%) | MCC  |
|---------|------------------|--------|--------|---------|------|
| m_b     | iRNA-m6A         | 79.32  | 76.90  | 78.75   | 0.58 |
|         | im6A-TS-CNN      | 81.50  | 75.85  | 78.67   | 0.5749 |
|         | our model        | **82.65** | **77.06** | **79.91** | **0.5991** |
| m_h     | iRNA-m6A         | 75.24  | 68.97  | 72.76   | 0.44 |
|         | im6A-TS-CNN      | 78.37  | 67.60  | 72.99   | 0.4633 |
|         | our model        | **81.56** | **70.64** | **76.10** | **0.5253** |
| m_k     | iRNA-m6A         | 82.60  | 77.31  | 79.98   | 0.60 |
|         | im6A-TS-CNN      | **81.00** | **80.46** | **80.46** | **0.6094** |
|         | our model        | **84.10** | **78.93** | **81.74** | **0.6361** |
| m_l     | iRNA-m6A         | 74.93  | 65.59  | 70.59   | 0.41 |
|         | im6A-TS-CNN      | 72.39  | 70.24  | 71.32   | 0.4288 |
|         | our model        | **76.95** | **70.87** | **73.92** | **0.4798** |
| m_t     | iRNA-m6A         | 78.14  | 70.02  | 74.40   | 0.48 |
|         | im6A-TS-CNN      | 75.21  | 75.61  | 75.41   | 0.5090 |
|         | our model        | **82.28** | **72.00** | **77.20** | **0.5474** |
| r_b     | iRNA-m6A         | 77.00  | 73.47  | 75.96   | 0.50 |
|         | im6A-TS-CNN      | 79.04  | 74.23  | 76.64   | 0.5379 |
|         | our model        | **81.25** | **75.22** | **78.27** | **0.5671** |
| r_k     | iRNA-m6A         | 82.46  | 80.05  | 81.78   | 0.63 |
|         | im6A-TS-CNN      | **84.15** | **82.46** | **82.46** | **0.6500** |
|         | our model        | **84.13** | **77.79** | **80.96** | **0.6602** |
| r_l     | iRNA-m6A         | 83.09  | 76.33  | 80.59   | 0.60 |
|         | im6A-TS-CNN      | 81.56  | 79.63  | 80.59   | 0.6126 |
|         | our model        | **83.43** | **81.64** | **82.46** | **0.6508** |

TABLE S3

Comparison of our model with im6A-TS-CNN and iRNA-m6A on independent test

| Species | Methods          | Sn (%) | Sp (%) | Acc (%) | MCC  |
|---------|------------------|--------|--------|---------|------|
| h_b     | iRNA-m6A         | 69.50  | 72.98  | 71.10   | 0.42 |
|         | im6A-TS-CNN      | 75.17  | 70.20  | 72.69   | 0.4543 |
|         | our model        | **81.82** | **74.62** | **74.62** | **0.4976** |
| h_k     | iRNA-m6A         | 77.13  | 78.42  | 77.76   | 0.56 |
|         | im6A-TS-CNN      | 79.95  | 78.53  | 79.24   | 0.5848 |
|         | our model        | **82.33** | **79.84** | **79.84** | **0.5975** |
| h_l     | iRNA-m6A         | 78.19  | 79.87  | 79.01   | 0.58 |
|         | im6A-TS-CNN      | **84.81** | **75.02** | **75.02** | **0.6012** |
|         | our model        | **84.13** | **77.79** | **80.96** | **0.6205** |
| m_b     | iRNA-m6A         | 77.20  | 79.41  | 78.26   | 0.57 |
|         | im6A-TS-CNN      | **86.22** | **70.74** | **78.48** | **0.5765** |
|         | our model        | **82.92** | **75.85** | **79.38** | **0.5891** |
| m_h     | iRNA-m6A         | 70.52  | 72.13  | 71.30   | 0.43 |
|         | im6A-TS-CNN      | 75.82  | 71.36  | 73.59   | 0.4723 |
|         | our model        | **80.36** | **75.34** | **75.34** | **0.5094** |
| m_k     | iRNA-m6A         | 78.37  | 80.32  | 79.31   | 0.59 |
|         | im6A-TS-CNN      | 80.52  | **81.00** | **80.76** | **0.6151** |
|         | our model        | **84.19** | **78.62** | **81.40** | **0.6290** |
| Species | Methods       | Sn (%) | Sp (%) | Acc (%) | MCC  |
|---------|---------------|--------|--------|---------|------|
| m_l     | iRNA-m6A      | 67.82  | 69.86  | 68.79   | 0.38 |
|         | im6A-TS-CNN   | 75.56  | 67.58  | 71.57   | 0.4328 |
|         | our model     | 75.34  | 70.72  | 73.03   | 0.4611 |
| m_t     | iRNA-m6A      | 72.19  | 75.08  | 73.54   | 0.47 |
|         | im6A-TS-CNN   | 83.45  | 68.87  | 76.16   | 0.5288 |
|         | our model     | 81.68  | 72.23  | 76.96   | 0.5415 |
| r_b     | iRNA-m6A      | 73.93  |        | 75.14   | 0.50 |
|         | im6A-TS-CNN   |        |        | 76.95   | 0.5391 |
|         | our model     | 79.71  | 75.29  | 77.50   | 0.5505 |
| r_k     | iRNA-m6A      | 80.18  | 82.77  | 81.42   | 0.63 |
|         | im6A-TS-CNN   |        |        | 82.72   | 0.6550 |
|         | our model     | 84.82  |        | 83.45   | 0.6692 |
| r_l     | iRNA-m6A      | 77.74  | 82.31  | 79.85   | 0.60 |
|         | im6A-TS-CNN   |        |        | 80.22   | 0.6067 |
|         | our model     | 84.17  | 78.83  | 81.50   | 0.6309 |

**TABLE S4**

The Bayesian optimization of parameters of the models for *Homo sapiens* and their corresponding AUC values

| Site | Iter | AUC  | BatchSize | Dropout | Filter1 | Filter2 | Pool_size |
|------|------|------|-----------|---------|---------|---------|-----------|
| h_b  | 1    | 0.8179 | 58.26    | 0.4733  | 36.93   | 18.32   | 4.421     |
|      | 2    | 0.8138 | 28.18    | 0.3217  | 49.46   | 59.49   | 1.396     |
|      | 3    | 0.8151 | 59.94    | 0.4248  | 53.48   | 52.05   | 1.087     |
|      | 4    | 0.8147 | 25.95    | 0.262   | 35.09   | 23.58   | 3.233     |
|      | 5    | 0.8083 | 37.67    | 0.7697  | 45.2    | 54.46   | 4.385     |
|      | 6    | 0.8148 | 59.45    | 0.404   | 40.12   | 16.06   | 1.833     |
|      | 7    | 0.8281 | 54.73    | 0.1725  | 35.69   | 20.81   | 5         |
|      | 8    | 0.812  | 54.37    | 0.3625  | 35.45   | 23.15   | 4.707     |
|      | 9    | 0.7931 | 57.46    | 0.8763  | 37.05   | 18.55   | 3.137     |
|      | 10   | 0.8198 | 41.53    | 0.3397  | 18.51   | 57.52   | 3.572     |
|      | 11   | 0.8185 | 58.46    | 0.5256  | 36.27   | 18.39   | 4.917     |
|      | 12   | 0.8166 | 42.22    | 0.6375  | 19      | 55.76   | 3.65      |
|      | 13   | 0.8099 | 40.6     | 0.1195  | 19.13   | 57.7    | 4.218     |
|      | 14   | 0.7949 | 42.41    | 0.8663  | 18.45   | 58.83   | 3.549     |
|      | 15   | 0.8108 | 37.71    | 0.1253  | 60.34   | 20.41   | 1.787     |
|      | 16   | 0.8083 | 26.62    | 0.7123  | 35.1    | 23.44   | 2.947     |
|      | 17   | 0.7872 | 58.62    | 0.8791  | 36.55   | 18.45   | 4.474     |
|      | 18   | 0.8067 | 47.71    | 0.7671  | 46.4    | 62.58   | 3.441     |
|      | 19   | 0.8004 | 56.99    | 0.2105  | 32.53   | 34.94   | 1.373     |
|      | 20   | 0.815  | 31.8     | 0.5697  | 42.38   | 37.04   | 4.057     |
|      | 21   | 0.8191 | 33.78    | 0.5886  | 20.38   | 35.46   | 1.81      |
|      | 22   | 0.8116 | 29.97    | 0.7187  | 34.7    | 44.49   | 4.749     |
|      | 23   | 0.8158 | 61.54    | 0.6018  | 37.22   | 17.1    | 3.574     |

(Continued)
Table S4 (continued).

| Site | Iter | AUC  | BatchSize | Dropout | Filter1 | Filter2 | Pool_size |
|------|------|------|-----------|---------|---------|---------|-----------|
| 24   | 0.8104 | 39.34 | 0.1123 | 18.12 | 26.62 | 4.678 |
| 25   | 0.808 | 51.52 | 0.7274 | 21.13 | 35.18 | 1.646 |
| 26   | 0.8146 | 30.16 | 0.7064 | 28.67 | 28.26 | 3.524 |
| 27   | 0.807 | 20.28 | 0.6955 | 63.51 | 60.14 | 1.745 |
| 28   | 0.8017 | 54.06 | 0.1049 | 20.94 | 20.32 | 2.206 |
| 29   | 0.7925 | 47.68 | 0.8692 | 48.22 | 43.16 | 4.589 |
| 30   | 0.7836 | 52.37 | 0.8856 | 26.76 | 51.14 | 4.636 |
| h_k  | 1   | 0.8865 | 27.62 | 0.1822 | 63.49 | 57.64 | 1.022 |
| 2    | 0.8862 | 17.25 | 0.8519 | 19.04 | 18.11 | 4.207 |
| 3    | 0.8852 | 30.91 | 0.8505 | 55.14 | 60.01 | 1.146 |
| 4    | 0.8957 | 24.91 | 0.4471 | 55.84 | 58.81 | 3.029 |
| 5    | 0.8892 | 62.59 | 0.6936 | 35.71 | 58.66 | 1.323 |
| 6    | 0.8892 | 36.86 | 0.8033 | 53.87 | 32.8 | 1.581 |
| 7    | 0.8959 | 29.23 | 0.4374 | 16.45 | 27.14 | 4.664 |
| 8    | 0.8877 | 25.2 | 0.8051 | 61.32 | 46.61 | 2.356 |
| 9    | 0.8898 | 21.21 | 0.1127 | 62.65 | 32.66 | 4.801 |
| 10   | 0.8932 | 26.14 | 0.2924 | 55.64 | 59.26 | 3.719 |
| 11   | 0.8881 | 28.12 | 0.7712 | 16.7 | 26.53 | 2.585 |
| 12   | 0.8935 | 29.44 | 0.6128 | 17.31 | 26.67 | 4.812 |
| 13   | 0.8901 | 24.43 | 0.2625 | 55.98 | 58.03 | 4.167 |
| 14   | 0.8927 | 23.66 | 0.6515 | 56.06 | 59.04 | 2.588 |
| 15   | 0.8923 | 31 | 0.7073 | 16.62 | 25.86 | 4.236 |
| 16   | 0.8917 | 25.56 | 0.1618 | 55.95 | 60.06 | 2.138 |
| 17   | 0.8923 | 25.66 | 0.7022 | 55.78 | 57.91 | 2.251 |
| 18   | 0.8805 | 24.92 | 0.9 | 54.85 | 59.18 | 3.011 |
| 19   | 0.8958 | 59.21 | 0.5495 | 16.44 | 31.61 | 4.712 |
| 20   | 0.8887 | 53.69 | 0.2379 | 52.76 | 47.04 | 1.124 |
| 21   | 0.8937 | 44.33 | 0.3335 | 33.72 | 46.8 | 1.279 |
| 22   | 0.8934 | 51.61 | 0.5804 | 50.05 | 62.51 | 4.767 |
| 23   | 0.8877 | 19.55 | 0.8194 | 34.65 | 57.58 | 4.131 |
| 24   | 0.8924 | 18.18 | 0.2217 | 18.58 | 33.29 | 3.279 |
| 25   | 0.8866 | 26.36 | 0.8189 | 55.52 | 59.83 | 4.239 |
| 26   | 0.8922 | 53.93 | 0.6832 | 27.44 | 45.65 | 2.076 |
| 27   | 0.8966 | 63.69 | 0.2681 | 32.11 | 57.99 | 3.167 |
| 28   | 0.8817 | 53.69 | 0.8958 | 36.72 | 24.32 | 1.576 |
| 29   | 0.8883 | 58.26 | 0.7604 | 32.55 | 48.2 | 2.216 |
| 30   | 0.8958 | 38.95 | 0.3098 | 31.45 | 25.63 | 3.056 |
| h_l  | 1   | 0.8916 | 53.6 | 0.311 | 31.46 | 22.42 | 2.128 |
| 2    | 0.8839 | 17.65 | 0.2256 | 17.22 | 56.2 | 2.226 |
| 3    | 0.8885 | 60.54 | 0.304 | 51.54 | 60.88 | 4.216 |
| 4    | 0.8955 | 52.81 | 0.4069 | 30 | 43.98 | 2.922 |
| 5    | 0.8864 | 34.16 | 0.2884 | 25.94 | 54.17 | 1.926 |
| 6    | 0.8938 | 64 | 0.5 | 24.79 | 40.7 | 5 |
| 7    | 0.8952 | 63.53 | 0.1774 | 23.88 | 39.39 | 4.683 |
| 8    | 0.8914 | 54.86 | 0.1 | 21.84 | 36.41 | 1.494 |
| 9    | 0.8889 | 57.41 | 0.1 | 31.49 | 37.73 | 4.523 |
| 10   | 0.8953 | 55.69 | 0.3392 | 26.53 | 46.24 | 2.447 |

(Continued)
| Site | Iter | AUC  | BatchSize | Dropout | Filter1 | Filter2 | Pool_size |
|------|------|------|-----------|---------|---------|---------|-----------|
| 11   | 0.8825 | 51.8 | 0.1       | 30.6    | 50.1    | 1       |
| 12   | 0.8939 | 54.38| 0.5       | 26.92   | 42.96   | 3.386   |
| 13   | 0.8878 | 57.71| 0.109     | 30.3    | 44.69   | 3.078   |
| 14   | 0.8951 | 51.12| 0.2503    | 26.74   | 44.41   | 4.487   |
| 15   | 0.8948 | 50.52| 0.5       | 28.24   | 42.28   | 1.554   |
| 16   | 0.8915 | 53.2 | 0.4991    | 22.75   | 45.03   | 1.885   |
| 17   | 0.8963 | 50.22| 0.5       | 30.3    | 42.07   | 5       |
| 18   | 0.8926 | 63.98| 0.4724    | 22.37   | 36.93   | 1.27    |
| 19   | 0.8932 | 48.51| 0.3468    | 32.98   | 39.46   | 1.988   |
| 20   | 0.8958 | 61.73| 0.3842    | 19.5    | 39.64   | 4.381   |
| 21   | 0.8881 | 60.1 | 0.1462    | 20.13   | 44.58   | 3.484   |
| 22   | 0.8929 | 64   | 0.1       | 18.74   | 36.75   | 5       |
| 23   | 0.8951 | 46.45| 0.1       | 28.38   | 42.1    | 5       |
| 24   | 0.897  | 45.1 | 0.4244    | 33.07   | 43.02   | 4.536   |
| 25   | 0.8958 | 41.83| 0.265     | 33.43   | 41.34   | 4.753   |
| 26   | 0.8931 | 43.31| 0.1812    | 37.88   | 43.07   | 4.694   |
| 27   | **0.8972** | **42.26** | **0.5** | **30.96** | **45.29** | **5** |
| 28   | 0.8935 | 43.09| 0.3186    | 30.57   | 43.63   | 2.289   |
| 29   | 0.8903 | 42.32| 0.2771    | 34.9    | 47.72   | 4.69    |
| 30   | 0.896  | 38.85| 0.5       | 29.44   | 43.49   | 5       |