DeepMethy: A deep learning model for protein methylation site prediction

Songye Gao*†‡, Cheng Xin*†‡, Sisi Ou†‡, Jinru Li†‡, Yanchun Liang‖§ Xiaohu Shi†‖§¶
*These authors contributed equally to this work
†College of Computer Science and Technology of Jilin University, Changchun 130012, China
‖School of Computer Science, Zhuhai college of Science and Technology, Zhuhai 519041, China
‡{sygao23, xincheng5520, ouss23, jinru23}@mails.jlu.edu.cn
§{ycliang, shixh}@jlu.edu.cn
¶To whom should be addressed

***Abstract*—Protein methylation is a crucial epigenetic modification that regulates protein stability, interactions and localization, and has a significant impact on biological processes such as gene expression and cellular signaling. The identification of methylation binding sites contributes to a better understanding of the molecular function of proteins. In this paper, a deep neural network based model is proposed to predict protein methylation, which is called by DeepMethy. The model utilizes two types of evolutionary features, namely BLOSUM score matrix and Position Specific Score Matrix (PSSM) as inputs, and extracts them at different sequence scales. The model employs CNN network as backbone and utilizes the attention mechanism for feature fusion. For the problem of sample imbalance, a weighted joint loss function is used for training. The experimental results show that DeepMethy has good performance, which is improved compared to existing SOTA methods, and the ablation experiments validate the advantages of the selected evolutionary features, as well as the effectiveness of the attention block and residual block of the model.**

***Index Terms*—Methylation, residual network, attention mechanism, BLOSUM62, Position Specific Score Matrix**

I. Introduction

Post-translational modification is a binding mechanism that regulates cellular functions through covalent and general enzymatic modifications [1]. It enhances the functional diversity of proteins [2] and plays an important role in regulating gene expression, cell division, cell signaling and other biological processes [3]. Protein methylation is an important form of PTM, which alters the structure and function of proteins by adding methyl groups to specific amino acid residues [4], and its role is not only limited to increasing the stability of proteins [5], but also includes regulating the activity of proteins [6], participating in signaling pathways [7], and interacting with other molecules in a variety of ways [8].

Experimental methods for identifying protein methylation are time-consuming, costly, and require specialized experimental techniques and equipment that do not meet practical needs. On the other hand, the gradual accumulation of methylation data has made it possible to perform methylation site prediction using computational methods. Computation methods can be categorized into traditional machine learning methods and deep learning methods. For traditional machine learning methods, Wei et al. [9] proposed a random forest-based model, MePredRF, to predict arginine and lysine methylation sites; Hu et al. [10] proposed a support vector machine based model, MeMo, and provided a web tool for arginine and lysine methylation site recognition; Abbas et al. [11] combined five different feature encoding techniques and used XGBoost as a classifier to predict the m5C methylation sites on RNA. Machine learning methods usually rely on manually selected features for model training, which can lead to biased feature selection and high computational costs [12].

With the rapid development of deep learning technology, researchers commonly use various deep learning models for methylation site prediction [13]. These methods are explored in two dimensions, one is how to represent the proteins and the other is what deep learning framework to build. For instance, by integrating CNN and bidirectional long-term and short-term memory networks (BiLSTM), Park et al. proposed a called DeepRMethylSite for predicting methylation sites [14]; Chen et al. applied ensemble learning to integrate five LSTM and CNN classifiers according to voting strategy, and developed DeepM6ASeq-EL model for methylation site recognition [15]. Khandelwal and Rout used Gated Recurrent Unit and CNN networks to extract sequential and spatial information from sequences and proposed DeepPRMS model for methylation site recognition [16]; Zhao et al. used one-hot encoding and CNN structure to generate a called CNNArginineMe model for arginine methylation site prediction [17]. Although the existing methods have made great progress, there is still room for improvement.

For protein methylation site identification, this paper proposes a called DeepMethy model that uses CNN networks as backbone supplemented with dense connected networks and residual structures. The model considers two evolutionary features, namely BLOSUM score matrix and Position Specific Score Matrix (PSSM), taking them as inputs. Moreover, the features are extracted at different sequence scales. To accommodate multi-modal features at multiple scales, we design a novel fusion mechanism that consists of the fusion of the same features at different sequence scales and the fusion of

Yanchun Liang's research is supported by NSFC (62372494) and the Guangdong Key Disciplines Project (2021ZDJS138) Projects. Xiaohu Shi's research is supported by NSFC (62272192).

different features at the same sequence scale. To reduce the effect of unbalanced samples, a weighted joint loss function is used for training. Experimental results show that the proposed DeepMethy model has superior performance compared to existing State-of-the-Art models.

II. Materials and methods

A. Overview of DeepMethy

As shown in Fig. 1, the proposed DeepMethy model utilizes the CNN networks as backbone, supplemented by densely connected and residual structures to recognize protein methylation sites. The model consists of four modules, namely input module, feature abstraction module, fusion module, fully connected network and output module. As a result of our research, we found that evolutionary features have a stronger discriminatory power in identifying protein methylation sites. Therefore, two evolutionary features are used as inputs to the model, namely the BLOSUM score matrix and the position-specific score matrix (PSSM), which are extracted within a window centered on the related amino acids. In order to fully capture the features at different distance scales from the center amino acid, different window scales are selected for feature extraction in Input Module. Both BLOSUM and PSSM features are two-dimensional matrices, with one dimension being the sequence length and the other the amino acid profile. Thus, 1d-convolution on both dimensions is performed on BLOSUM and PSSM features separately in the Feature Extraction Module. After the convolution block, followed by dense block and residual block for further feature extraction. Then the outputs of Feature Extraction Module are the representations of multi-module features after two 1d-convolution on different sequence scales, which are fused on the Fusion Module. The multi-channel feature representations of the same feature after 1d-convolution in the same dimension are fused by the attention block, and then the representations of different features after 1d-convolution in the same dimension are concatenated for fusion. Finally, the fused feature representation of the Fusion Module enters the Fully Connected Network and Output Module, which is firstly flattened, then goes through a fully connected layer, and finally passes through the softmax layer to get the possibility of methylation.

B. Data collection and preprocessing

In this paper, we focus our experiments on arginine methylation site recognition. We searched 1785 arginine methylated protein sequences in Swiss-Prot [18] and precisely identified arginine methylated sites based on the PTM (post-translational modification) information, which are set as positive samples. The remaining arginine sites in the above sequences, as well as those in other protein sequences, are set as negative samples. Protein sequences of length 41 are intercepted centered on the confirmed arginine methylation sites, and the missing residues complement "0" when the peptide sequence is not long enough. Considering that the close distance between some arginine sites may lead to high similarity between the intercepted sequences, we applied the CD-HIT tool [19] to remove protein sequences with homology greater than 30%, and finally obtained a protein sequence dataset containing 74,178 arginine sites, 3,199 of which are methylated, being the positive samples, and the other 70,949 are the negative samples. They are divided into a training set, which contains 71,618 samples, of which 2559 are positive and the remaining 69,059 are negative, and a test set, which contains 2,560 samples, of which 640 are positive and the remaining 1,920 are negative. In order to adequately capture features at different distance scales from the center arginine, we choose three window scales of $L = 41, 25, 11$ for feature extraction.

C. Input Module

According to our experience, evolutionary features are found to have better performance in the task of protein methylation site identification, so in this paper, two evolutionary features are chosen as inputs to the model, namely Blocks Substitution Matrix (BLOSUM) [20] and Position Specific Score Matrix (PSSM) [21]. BLOSUM are computed from amino acid substitutions observed in highly conserved ungapped alignment regions of evolutionary divergent proteins. They are a series of matrices, a BLOSUM-X matrix is derived from alignments such that no sequence is more than X% identical to any other sequence in the alignment. In this paper, BLOSUM62 is used for representation. PSSM represents the probability of occurrence of any amino acids at each position of a given sequence, extracting by using position-specific iterative blast (PSI-BLAST) program. Thus, for a protein sequence of length l, its BLOSUM62 is represented as a $21 \times l$ matrix (21 for the 20 basic and unknown amino acids), while its PSSM is represented as a $20 \times l$ matrix (20 for the 20 basic amino acids).

In order to finely capture features at different distance scales from the center amino acid, 3 different window scales ($L = 41, 25, 11$) are selected in the input module for feature extraction.

Thus, for a protein sequence of length l, its BLOSUM62 is represented as a $21 \times l$ matrix (21 represents the 20 basic types of amino acids and unknown amino acids), and its PSSM is represented as a $20 \times l$ matrix. For the later need of 1d-convolution from the sequence dimension and the amino acid spectrum dimension, respectively, the raw features and their transpositions for each of the three windows of BLOUM62 and PSSM are fed into the Feature Extraction Module, respectively.

D. Feature Abstraction Module

As mentioned above, the original features of each of the 3 sets of windows of BLOUM62 and PSSM and their transpositions totaling 12 sets of features are input to the Feature Extraction Module, respectively. Inside this module, the 12 groups of features are processed in parallel with the same network structure, successively passing through convolutional blocks, densely connected blocks, activation layers, and residual blocks (See Feature Extraction Module in Fig. 1. They are described in detail below.

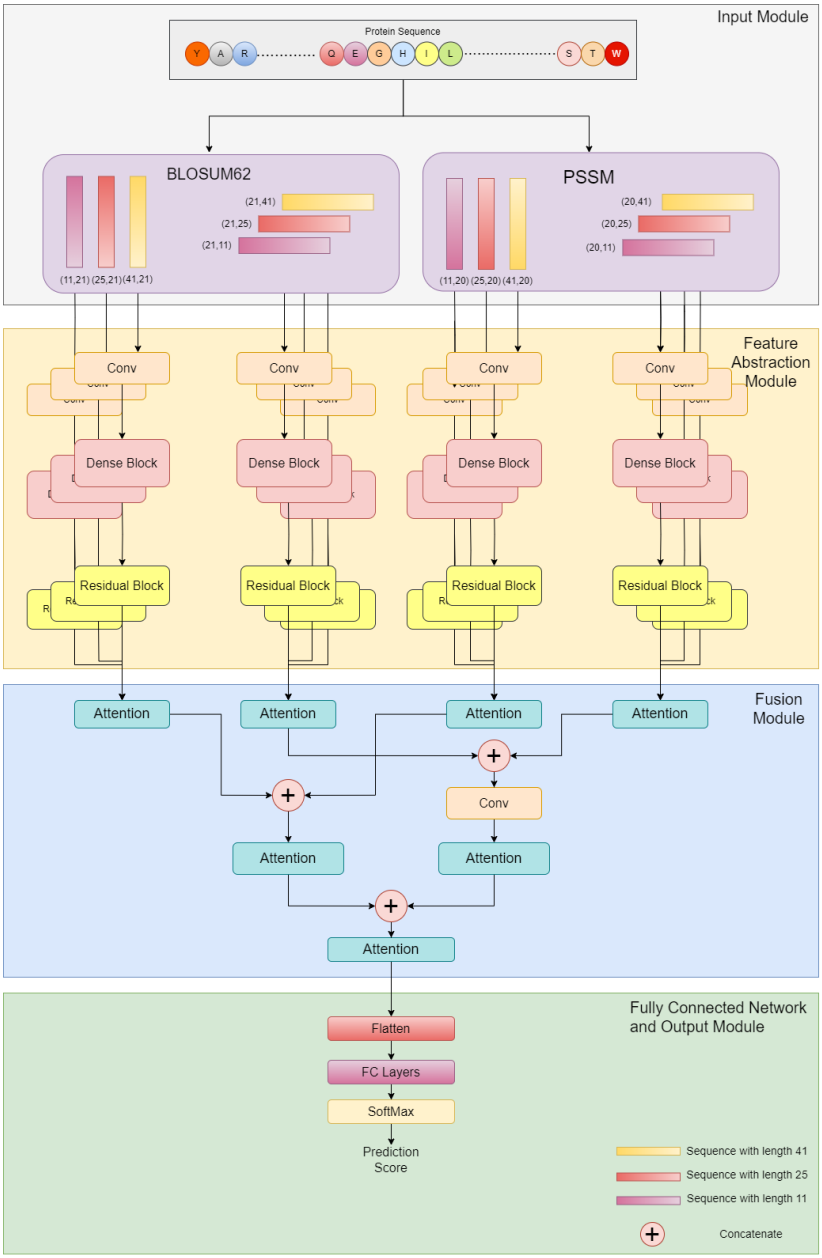

Fig. 1: Framework of DeepMethy

1) 1d-CNN block: Both BLOSUM and PSSM features are two-dimensional matrices, where one dimension represents the sequence length and the other the amino acid profile. Therefore, 1d-convolution is performed on both dimensions separately to extract the features in both dimensions. The 1d-convolution operation uses 32 kernels, each of which is of a size of L (the length of whole window), and carries out nonlinear transformation through the ReLU activation function. In addition, the weight of the convolution kernel is constrained by L2 regularization to control the complexity of the model and prevent overfitting. Through the convolution operation, two types of features, namely the features along the sequence and along the amino acid profile are captured.

2) Dense-Block: The abstracted 12 sets of features extracted from 1d-CNN block are input into the sense block, each of which passes through a dense unit in parallel. Each dense unit contains C convolutional layers, the structure of which is shown in Fig. 2. For the i-th layer, its input is the concatenation of the inputs of all previous layers, and the output of which is

$$\mathcal{D}_i^k = \alpha_i^k \left(W_i^k \left[\mathcal{D}_0^k, \mathcal{D}_1^k, \ldots, \mathcal{D}_{i-1}^k \right] + b_i^k \right) \quad 1 \le i \le C, 1 \le k \le K \tag{1}$$

where k is channel index for different window scale, i the layer index, C the layer number (5 in our work), K the window scale channel number (3 in our work), $\mathcal{D}_i^k$ the feature map generated by the i-th layer of the k-th channel ($\mathcal{D}_0^k$ means

the input of this dense unit), W_i^k and b_i^k are the weight and the bias correspondingly, and α_k is the active function (ReLU function is selected in our work), respectively.

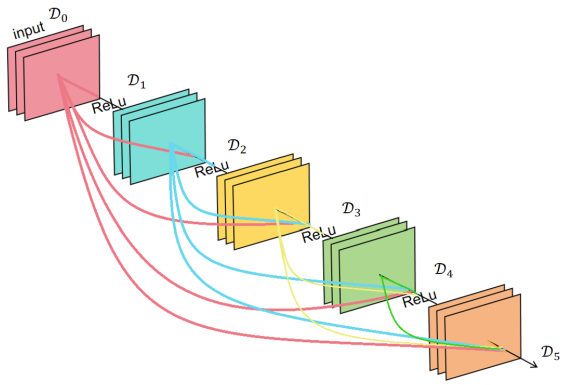

Fig. 2: Structure of Dense Unit

3) Residual-Block: The output of dense block corresponding to 12 sets of features generated from the 3 sets of windows of BLOUM62 and PSSM and their transpositions. They also pass through the residual block in parallel [22]. Each residual unit including two convolutional layers, and the output of the unit is :

$$R_k = f\left(W_{k2} f\left(W_{k1}\mathcal{D}_k + b_{r1}\right) + b_{r2}\right) \tag{2}$$

where R_k and $\mathcal{D}_k$ are the outputs of residual unit and dense unit corresponding to the k-th window scale channel, W_{k1} and W_{k2} the weight matrices of the first and second layer, b_{r1} and b_{r2} the bias correspondingly, and $f()$ is the active function (such as ReLU).

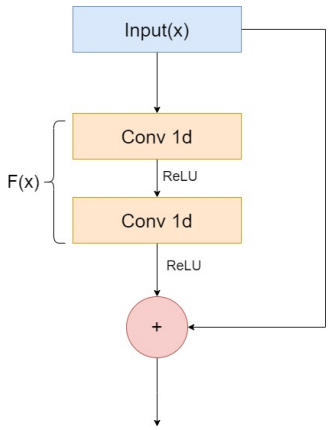

Fig. 3: Structure of Residual Unit

E. Fusion Module

The fusion module first fuses the same features with the same encoding mode on different window scales by using attention blocks, then concatenate the same encoding modes of different features and fuses them again by using attention blocks, and finally concatenate and utilizes attention blocks again to obtain the final fused feature. The attention block applies multi-head attention mechanism, the structure of which is shown as Fig. 4. The details of the attention block are described as follow:

Attention Block: Attention blocks are used at three levels in the Fused Module: 1). fusion of feature maps at different window scales for the same features extracted in the same dimension as the output of the Feature Extraction Module (the first layer of the Fused Module); 2) fusion of feature maps for different features in the same dimension (either the sequence dimension or the amino acid profile dimension) (the second layer of the Fused Module); and 3) fusion of feature maps obtained from the sequence dimension and the amino acid profile dimension (the third layer of the Fused Module).
Each attention block uses a multi-head attention mechanism. Denote the input of an attention block as R, then for each head the linear transformations are first computed as follows:

$$Q_h = W_h^q R \tag{3}$$

$$K_h = W_h^k R \tag{4}$$

$$V_h = W_h^v R \tag{5}$$

where h is the index number of head, Q_h, K_h, and V_h are the query matrix, key matrix and value matrix, and W_h^q, W_h^k, and W_h^v are the weight matrices correspondingly. Next, the dot product of the query and key is calculated, and the resulting attention weights α_h are normalized via the softmax function:

$$\alpha_h = soft\max\left(\frac{q_h k_h^T}{\sqrt{d}}\right) \tag{6}$$

where d is the dimension of the keys. The attention weights are then multiplied by the corresponding value vectors to obtain the weighted output O_h:

$$O_h = \alpha_h V_h \tag{7}$$

Finally, the outputs from all heads are concatenated and passed through a linear transformation to produce the final attention output O:

$$O = W^o\left[O_1; O_2; \ldots; O_H\right] \tag{8}$$

where W^o is the weight matrix for the output, and H is the number of attention heads.

F. Full Connected Network and Output Module

1) Structure of the module: In the above way, multiple feature *maps* are connected and then converted into an one-dimensional vector $b_f \in \mathbb{R}^d$ through a flatten layer. Then, a fully connected neural network is applied to transform it as:

$$f_c = b_f W^f \tag{9}$$

where W_f is the trainable weight matrix. Finally, it is used as an input to the softmax layer to get the likelihood of the target amino acid being methylated.

$$P\left(y = 1 | f_c\right) = \frac{1}{1 + e^{-f_c \times W_c}} \tag{10}$$

where W_c is the trainable weight matrix.

TABLE I: Comparison results with other embedding methods

	Accuracy	Precision	Recall	F1 Score	MCC
BLOSUM+PSSM	**0.9629**	**0.9633**	**0.9629**	**0.9622**	**0.8991**
BLOSUM	0.9574	0.9581	0.9574	0.9564	0.8849
PSSM	0.7528	0.5666	0.7528	0.6466	0.5500
ProteinBERT	0.9343	0.9298	0.9342	0.9067	0.7525
one-hot	0.9535	0.9548	0.9535	0.9522	0.8744

TABLE II: Comparison results with other deep learning models

	Accuracy	Precision	Recall	F1 Score	MCC
DeepMethy	**0.9629**	**0.9633**	**0.9629**	**0.9622**	**0.8991**
DeepPRMS	0.9574	0.9594	0.9574	0.9561	0.8858
CNNArginineMe	0.8027	0.8238	0.8027	0.7554	0.3995
DeepRMethylSite	0.9010	0.9014	0.8970	0.8964	0.7300

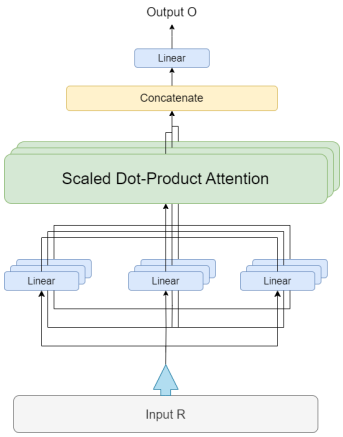

Fig. 4: Structure of Attention Block

2) Weighted joint loss function: In view of the imbalance of data sets, DeepMethy uses the weighted information entropy method to evaluate the accuracy of classification prediction results. By setting the weight of positive and negative samples, the imbalance of data set can be alleviated.
First, we define the weight of each sample:

$$W_j = \frac{\sum_{j=1}^{N} y_{true}^j}{\sum_{j=1}^{N}\left(1 - y_{true}^j\right) + \varepsilon} \tag{11}$$

where N represents the number of samples and y_{true}^j represents the real label of the j-th sample. In order to avoid 0 as a divisor, we set the ε to represent infinitesimal.
The standard cross entropy is:

$$\begin{aligned} L \;=\; -\frac{1}{N}\sum_{j=1}^{N}\Big(y_{true}^j \log\left(y_{pred}^j = 1|x^j\right) \\ + \left(1 - y_{true}^j\right)\log\left(y_{pred}^j = 0|x^j\right)\Big) \end{aligned} \tag{12}$$

where y_{pred}^j represents the prediction label of the j-th sample. When considering the imbalance of different classes of samples, it could be rewritten as:

$$\begin{aligned} L_c \;=\; \frac{1}{N}\sum_{j=1}^{N} W_j \times \Big(-\Big(y_{true}^j \log\left(y_{pred}^j = 1|x^j\right) \\ + \left(1 - y_{true}^j\right)\log\left(y_{pred}^j = 0|x^j\right)\Big)\Big) \end{aligned} \tag{13}$$

III. Experiments and Results

A. Evaluation measurements

We assess the prediction performance of different algorithms by evaluating metrics such as accuracy (ACC), specificity (SP), sensitivity (SN), F1 score, and Matthews correlation coefficient (MCC). These evaluation metrics have been used extensively in previous bioinformatics work [23] [24].

$$Acc = \frac{TP + TN}{TP + TN + FP + FN} \tag{14}$$

$$Sp = \frac{TN}{TP + FP} \tag{15}$$

$$\text{Re}call = Sn = \frac{TP}{TP + FN} \tag{16}$$

$$F1 = \frac{2 \times Pre \times Sn}{Pre + Sn} \tag{17}$$

$$MCC = \frac{TP \times TN - FP \times FN}{\sqrt{(TP + FN) \times (FP + TP) \times (TN + FN) \times (TN + FP)}} \tag{18}$$

where, Tp refers to the number of positive samples correctly classified by the predictor, while Tn represents the number of negative samples correctly identified. Fp and Fn denote the counts of positive and negative samples that have been incorrectly classified, respectively. Consequently, Sn, or sensitivity, is the percentage of true positive samples correctly classified by the predictor, and similarly, Sp, or specificity, is the percentage of true negative samples correctly identified. Precision, denoted as Pre, is the ratio of true positives to all predicted positives, reflecting the accuracy of the predictor in generating true positive samples. The Matthews Correlation Coefficient (MCC) is a measure of the balance between the

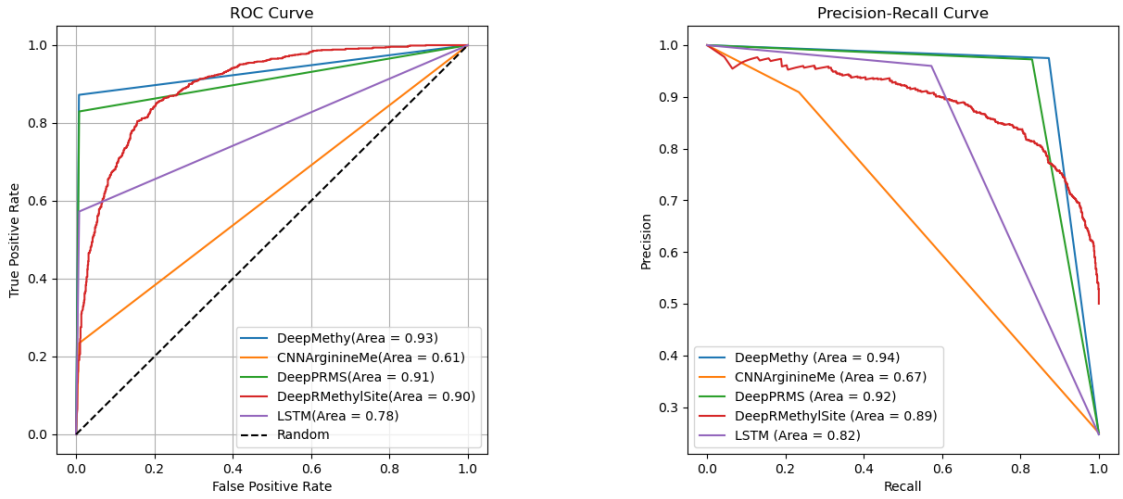

Fig. 5: ROC and PR curve for the different deep learning-based models.

TABLE III: Results of the test data in ablation experiments

	Accuracy	Precision	Recall	F1 Score	MCC
DeepMethy	**0.9629**	**0.9633**	**0.9629**	**0.9622**	**0.8991**
-Evolutionary information	0.7961	0.8308	0.7961	0.7401	0.3767
-Attention	0.9621	0.9624	0.9621	0.9615	0.8969
-Residual	0.9310	0.9354	0.9310	0.9272	0.8115

quality of positive and negative predictions. The $F1$ score is an indicator that takes into account both the accuracy and recall of the predictor. Additionally, we employ the Receiver Operating Characteristic (ROC) curve and the Area Under the Curve (AUC) to assess overall performance. A ROC curve closer to the top-left corner and an AUC value closer to 1 indicate better overall performance [25].

B. Comparison with other embedding methods

Based on the arginine methylation dataset, we systematically investigated the effects of different embedding methods in the DeepMethy framework. We conducted a comparative analysis between the BLOSUM62+PSSM encoding, employed by the DeepMethy model, and several alternative encoding techniques, including One-Hot, BLOSUM62, PSSM, and ProteinBert [26].

To rigorously quantify the importance of BLOSUM62+PSSM coding to the DeepMethy model, we compare the results of different coding methods on data sets. The experimental results are shown in the Table I, the results consistently indicate that the BLOSUM62+PSSM encoding method yields superior performance compared to other encoding strategies. The encoding approach introduced in this study not only incorporates the evolutionary information embedded within the amino acid sequence but also leverages the PSSM matrix, which encapsulates the conserved tendencies of amino acids at specific positions throughout the evolutionary process. This combined encoding framework thus provides critical insights into the structural and functional characteristics of proteins, offering a robust enhancement to the predictive capabilities of the model.

C. Comparison with existing methods

In this section, to evaluate the performance of DeepMethy, we use independent test data to compare DeepMethy with several existing tools for predicting general methylation sites. The methylation prediction of R site is compared with several well-known deep learning models, including DeepPRMS, CNNArginineMe, and DeepRMethylSite. The comparison results of the methylation prediction tools based on the arginine methylation independent test set are shown in Table II.

Overall, DeepMethy achieves higher performance than the other four predictors. In all indicators, the performance of DeepMethy is better than the other four deep learning prediction models.

D. Ablation experiment and visualization for features

To investigate the impact of coding schemes, attention mechanism, residual network and other components on the results, we carried out ablation experiments. 1) the coding representation model only uses sequence information, excluding protein evolution-related data; 2) -Attention indicates that the attention mechanism has been removed from the model. 3) -Residual indicates that the residual network structure is

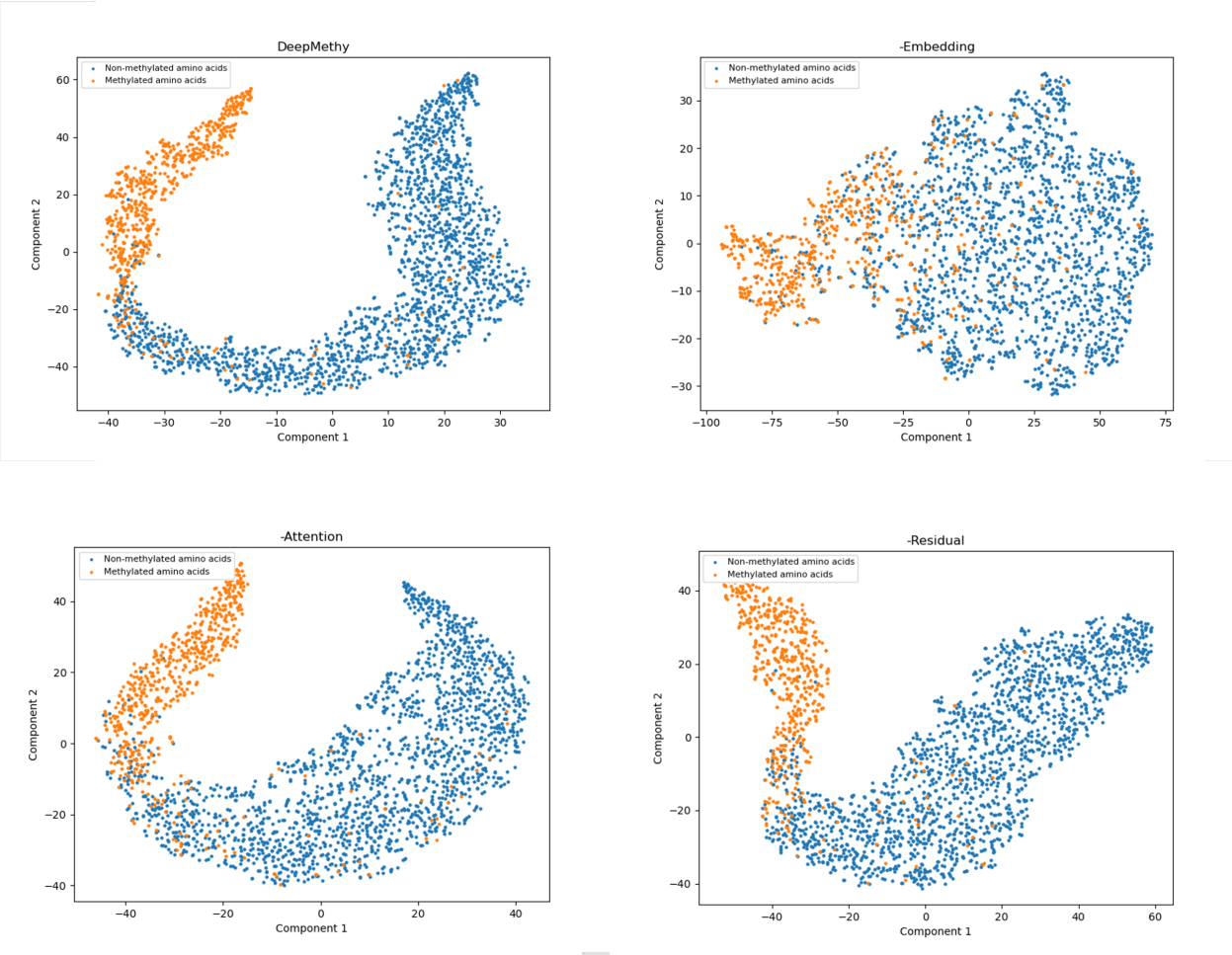

Fig. 6: Visualization of features between DeepMethy and different component ablation experiments.

removed from the model.

To assess the contribution of each component, we applied the t-SNE [27] algorithm to visualize the outcomes of the ablation experiments. As illustrated in Figure 6 and detailed in Table III, DeepMethy effectively distinguishes between positive and negative samples, outperforming all other ablation conditions. Among them, the protein evolution information was found to have the most significant effect on the performance of the model. After removing the evolution information, the MCC of the model is reduced from 0.8991 to 0.3767. These findings underscore that sequence information alone is insufficient to accurately capture the corresponding biological functions. Conversely, the utilization of residual network structure can increase MCC by 8.76%, demonstrating the benefit of residual network. In addition, the integration of attention mechanism helps the model to focus on useful information and helps to improve the predictive ability of the model.

IV. Discussion and Conclusions

In recent years, protein methylation, as an important post-translational modification (PTM), has occupied an important position in fields such as biological research and disease prevention. However, experimental validation of methylation sites is both time-consuming and expensive, thus computationally based prediction methods are of great importance. To address this problem, this paper proposes a deep learning model for protein methylation site prediction, called by DeepMethy. Considering the effectiveness of evolutionary information on this task, BLOSUM62 and PSSM matrices are chosen to encode proteins. Meanwhile, in order to capture the specificity attribute of the distance from the central amino acid position, multi-channel is utilized for feature extraction at different window scales. DeepMethy takes convolutional network as the backbone network, and utilizes dense connection and residual connection structure. An important innovation of the model is the repeated use of the attention module to fuse the extracted feature maps of the same features at different window scales, and of different features in the sequence dimension or amino acid profile dimension, to fully carry out the flow of methylation information. Comparison results with existing methods demonstrate the better performance of our proposed model, and the ablation experimental results show

the validity of the evolutionary features, the attention block and the residual block used in the model.

The strengths of our method are highlighted in the following aspects:

- Two evolutionary protein features, namely the BLOSUM62 scoring matrix and the position-specific scoring matrix (PSSM), are used as protein representation, better accommodated to the task of protein methylation site prediction.
- Apply multiple channels to capture features under different window scales of amino acids in the center, thus to improve the extraction performance of position-sensitive features.
- Through the iterative application of the attention block, the multi-scale features under different window scales, under the sequence dimension and the amino acid spectrum dimension, and the final extracted features are fully fused to strengthen the flow mechanism of methylation information.

In the future, the model could be further improved in the following aspects. First, as the deep learning approach is still a black box, it needs to be further enhanced in terms of biological interpretation. In addition, some latest deep learning models are expected to further improve the performance of the algorithm.

REFERENCES

[1] Q. Zhong, X. N. Xiao, Y. J. Qiu, Z. Q. Xu, C. Y. Chen, B. C. Chong, X. J. Zhao, S. Hai, S. Q. Li, and Z. M. An et al. Protein posttranslational modifications in health and diseases: Functions, regulatory mechanisms, and therapeutic implications. *MedComm*, 4(3): e261, 2023.

[2] L. H. Zhai, K. F. Chen, B. B. Hao, and M. J. Tan. Proteomic characterization of post-translational modifications in drug discovery. *Acta Pharmacologica Sinica*, 43(12): 3112–3129, 2022.

[3] Y. C. Wang, S. E. Peterson, and J. F. Loring. Protein post-translational modifications and regulation of pluripotency in human stem cells. *Cell research*, 24(2): 143–160, 2014.

[4] S. Cristina, S. W. Jana, G. Fernando, H. Begoña, L. Vanesa, P. Patricia, Z. Eduardo, X. E. Pilar, O. Sagrario, and M. Diego et al. Decoding protein methylation function with thermal stability analysis. *Nature Communications*, 14(1): 3016, 2023.

[5] E. R. Gibney and C. M. Nolan. Epigenetics and gene expression. *Heredity*, 105(1): 4–13, 2010.

[6] K. K. Biggar and S. S. Li. Non-histone protein methylation as a regulator of cellular signalling and function. *Nature reviews Molecular cell biology*, 16(1): 5–17, 2015.

[7] H. H. Wei, X. J. Fan, Y. Hu, X. X. Tian, M. Guo, M. W. Mao, Z. Y. Fang, P. Wu, S. X. Gao, and C. Peng et al. A systematic survey of prmt interactomes reveals the key roles of arginine methylation in the global control of rna splicing and translation. *Science Bulletin*, 66(13): 1342–1357, 2021.

[8] Y. Y. Li and T. O. Tollefsbol. Dna methylation detection: bisulfite genomic sequencing analysis. *Epigenetics protocols*, 791: 11–21, 2011.

[9] L. Y. Wei, P. W. Xing, G. T. Shi, Z. L. Ji, and Q. Zou. Fast prediction of protein methylation sites using a sequence-based feature selection technique. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 16(4): 1264–1273, 2017.

[10] H. Chen, Y. Xue, N. Huang, X. B. Yao, and Z. R. Sun. Memo: a web tool for prediction of protein methylation modifications. *Nucleic Acids Research*, 34: W249–W253, 2006.

[11] Z. Abbas, M. Rehman, H. Tayara, Q. Zou, and K. T. Chong. Xgboost framework with feature selection for the prediction of rna n5-methylcytosine sites. *Molecular Therapy*, 31(8): 2543–2551, 2023.

[12] J. Cai, J. W. Luo, S. L. Wang, and S. Yang. Feature selection in machine learning: A new perspective. *Neurocomputing*, 300: 70–79, 2018.

[13] Y. Z. Gao and Y. Yuan. Feature-less end-to-end nested term extraction. In *Natural Language Processing and Chinese Computing*, pages 607–616. Springer, 2019.

[14] C. Park, J. Ha, and S. Park. Prediction of alzheimer's disease based on deep neural network by integrating gene expression and dna methylation dataset. *Expert Systems with Applications*, 140: 112873, 2020.

[15] J. T. Chen, Q. Zou, and J. Li. Deepm6aseq-el: prediction of human n6-methyladenosine (m 6 a) sites with lstm and ensemble learning. *Frontiers of Computer Science*, 16: 1–7, 2022.

[16] M. Khandelwal and R. R. Kumar. Deepprms: advanced deep learning model to predict protein arginine methylation sites. *Briefings in Functional Genomics*, 4: 4, 2024.

[17] J. J. Zhao, H. Q. Jiang, G. Y. Zou, Q. Lin, Q. Wang, J. Liu, and L. N. Ma. Cnnargininene: A cnn structure for training models for predicting arginine methylation sites based on the one-hot encoding of peptide sequence. *Frontiers in Genetics*, 13: 1036862, 2022.

[18] B. Boeckmann, A. Bairoch, R. Apweiler, M. C. Blatter, A. Estreicher, E. Gasteiger, M. J. Martin, K. Michoud, C. O'Donovan, and I. Phan et al. The swiss-prot protein knowledgebase and its supplement trembl in 2003. *Nucleic acids research*, 31(1): 365–370, 2003.

[19] L. M. Fu, B. F. Niu, Z. W. Zhu, S. T. Wu, and W. Z. Li. Cd-hit: accelerated for clustering the next-generation sequencing data. *Bioinformatics*, 28(23): 3150–3152, 2012.

[20] S. Henikoff and J. G. Henikoff. Amino acid substitution matrices from protein blocks. *Proceedings of the National Academy of Sciences*, 89(22): 10915–10919, 1992.

[21] S. Ahmad and A. Sarai. Pssm-based prediction of dna binding sites in proteins. *BMC bioinformatics*, 6: 1–6, 2005.

[22] K. M. He, X. Y. Zhang, S. Q. Ren, and J. Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[23] Y. J. Jang, Q. Q. Qin, S. Y. Huang, A. T. J. Peter, X. M. Ding, and B. Kornmann. Accurate prediction of protein function using statistics-informed graph networks. *Nature Communications*, 15(1): 6601, 2024.

[24] J. Jumper, R. Evans, A. Pritzel, T. Green, M. Figurnov, O. Ronneberger, K. Tunyasuvunakool, R. Bates, A. Dek, and A. Potapenko et al. Highly accurate protein structure prediction with alphafold. *nature*, 596(7873): 583–589, 2021.

[25] Y. Li, Z. Wang, L. P. Li, Z. H. You, W. Z. Huang, X. K. Zhan, and Y. B. Wang. Robust and accurate prediction of protein–protein interactions by exploiting evolutionary information. *Scientific Reports*, 11(1): 16910, 2021.

[26] Y. Q. Zhang, G. C. Zhu, K. W. Li, F. Li, L. Huang, M. Y. Duan, and F. F. Zhou. Hlab: learning the bilstm features from the protbert-encoded proteins for the class i hla-peptide binding prediction. *Briefings in Bioinformatics*, 23(5): bbac173, 2022.

[27] M. L. Van and G. Hinton. Visualizing data using t-sne. *Journal of machine learning research*, 9(11), 2008.