Contents lists available at ScienceDirect

Computers in Biology and Medicine

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

DE-MHAIPs: Identification of SARS-CoV-2 phosphorylation sites based on differential evolution multi-feature learning and multi-head attention mechanism

Minghui Wang [a], Lu Yan [a], Jihua Jia [a], Jiali Lai [a], Hongyan Zhou [a,**], Bin Yu [b,c,*]

[a] College of Mathematics and Physics, Qingdao University of Science and Technology, Qingdao, 266061, China
[b] College of Information Science and Technology, School of Data Science, Qingdao University of Science and Technology, Qingdao, 266061, China
[c] School of Data Science, University of Science and Technology of China, Hefei, 230027, China

ARTICLE INFO

Keywords:
SARS-CoV-2 phosphorylation sites
Multi-view feature fusion
Differential evolution algorithm
Group LASSO
Multi-head attention mechanism

ABSTRACT

The rapid spread of severe acute respiratory syndrome coronavirus 2 (SARS-CoV-2) around the world affects the normal lives of people all over the world. The computational methods can be used to accurately identify SARS-CoV-2 phosphorylation sites. In this paper, a new prediction model of SARS-CoV-2 phosphorylation sites, called DE-MHAIPs, is proposed. First, we use six feature extraction methods to extract protein sequence information from different perspectives. For the first time, we use a differential evolution (DE) algorithm to learn individual feature weights and fuse multi-information in a weighted combination. Next, Group LASSO is used to select a subset of good features. Then, the important protein information is given higher weight through multi-head attention. After that, the processed data is fed into long short-term memory network (LSTM) to further enhance model's ability to learn features. Finally, the data from LSTM are input into fully connected neural network (FCN) to predict SARS-CoV-2 phosphorylation sites. The AUC values of the S/T and Y datasets under 5-fold cross-validation reach 91.98% and 98.32%, respectively. The AUC values of the two datasets on the independent test set reach 91.72% and 97.78%, respectively. The experimental results show that the DE-MHAIPs method exhibits excellent predictive ability compared with other methods.

1. Introduction

The coronavirus Virus Disease 2019 (COVID-19) is extremely contagious and pathogenic, and it has spread rapidly around the world and caused an urgent health and socioeconomic crisis [1]. To date, there is no universally proven fully effective treatment. Although various vaccine candidates against SARS-CoV-2 are already available and have some effect on the SARS-CoV-2 currently circulating around the world, they may also fail to control the disease caused by SARS-CoV-2 [2]. In order to develop therapeutic strategies to combat SARS-CoV-2, understanding the interaction between the virus and the host is crucial for developing drugs against the coronavirus [3].

Proteomics can reveal the characteristics of SARS-CoV-2 infected cells, help people understand the molecular mechanisms of host cell regulation after SARS-CoV-2 infection, and identify drugs that inhibit the replication of the coronavirus [4]. Hekman et al. [5] explored the alveolar pathway and process of infection with SARS-CoV-2, combined primary human-like AT2 cells with phosphoproteomic analysis, and demonstrated differential host responses to infection of alveolar epithelial cells. Bouhaddou et al. [6] presented a quantitative mass spectrometry-based phosphoproteomics investigation of infection with SARS-CoV-2 in Vero E6 cells, and revealed the mechanism of action of phosphorylation on host and viral proteins. Klann et al. [7] studied the signaling of host cells infected with SARS-CoV-2 and found that the central pathway required for SARS-CoV-2 replication is GFR signaling, which led to a new approach for SARS-CoV-2 therapeutic strategies. Recently, Tung et al. [8] performed phosphorylation analysis of SARS-CoV-2 to better understand the mutations at phosphorylation sites. In verifying a large number of candidate phosphorylation sites, traditional experimental methods are laborious. Therefore, computational methods have become a better choice for identifying SARS-CoV-2 phosphorylation sites.

* Corresponding author. School of Data Science, University of Science and Technology of China, Hefei, 230027, China.
** Corresponding author. College of Mathematics and Physics, Qingdao University of Science and Technology, Qingdao, 266061, China.
E-mail addresses: zhouhongyan@qust.edu.cn (H. Zhou), yubin@qust.edu.cn (B. Yu).

https://doi.org/10.1016/j.compbiomed.2023.106935
Received 20 January 2023; Received in revised form 12 March 2023; Accepted 13 April 2023
Available online 14 April 2023

So far, many researchers have proposed many effective predictive models to identify phosphorylation sites. For example, Dou et al. [9] used SVM as a classifier combined with a sequence scoring function for identifying phosphorylation sites. Li et al. [10] predicted phosphorylation sites by combining multiple sequence scoring functions and logistic regression. Based on RF, Dang et al. [11] developed a phosphorylation site prediction tool called SKIPHOS. Based on CNN and a two-dimensional attention mechanism, Wang et al. [12] took raw sequence data as input and proposed a prediction model Musite-Deep2017 to predict phosphorylation sites. Based on LSTM, Ahmed et al. [13] proposed a phosphorylation prediction model, DeepPPSite, which extracted protein features by fusing PSPM, IPC and EGBW methods to predict the phosphorylation sites. Guo et al. [14] designed a deep neural network based on global-local information and proposed a model called DeepPSP for identifying phosphorylation sites. Recently, based on the CNN-LSTM network architecture, lv et al. [15] proposed a prediction model, DeepIPs, which employs word embedding methods to extract protein feature information. These deep learning methods have been shown to outperform previous traditional machine learning methods.

Although there are already many models for identifying SARS-CoV-2 phosphorylation sites, further improvements and breakthroughs can be made in terms of prediction accuracy. Inspired by this, we propose a new model, DE-MHAIPs, to predict the phosphorylation sites of SARS-CoV-2. Firstly, we extract sequences of proteins from different perspectives using dipeptide deviation from expected mean (DDE), pseudo amino acid composition (PseAAC), amino acid property coding (AAindex), composition, transition and distribution (CTD), enhanced amino acid composition (EAAC) and BLOSUM62 to get the original feature vector and use a DE algorithm to learn individual feature weights, and fuse these features through weighted combinations. Then, Group LASSO is used to screen optimal feature subset, and we input the screened date into the deep learning model to predict the phosphorylation site of SARS-CoV-2. The AUC values and AUPR values of the S/T and Y datasets under the 5-fold cross-validation achieve relatively satisfactory results. The experimental results show that the DE-MHAIPs model can better identify the phosphorylation sites of SARS-CoV-2.

2. Materials and methods

2.1. Datasets

Choosing an objective and symbolic data not only allows a fairer evaluation of the characteristics of the prediction model in this study, but also facilitates the comparison of the prediction results with other prediction results. The dataset constructed by Lv et al. [15], which is a collection of experimentally validated phosphorylation sites in human A549 cells infected with SARS-CoV-2 [16]. To reduce sequence redundancy in phosphorylated proteins, and the CD-HIT [17] was used to eliminate sequences with a protein sequence similarity greater than 30%. The processed sequence was truncated into a 33-residue sequence fragment with either S/T or Y at the residue center. If the central S/T or Y of the fragment is phosphorylated, it is defined as a positive sample, otherwise, it is defined as a negative sample. In summary, a total of 10,774 samples were obtained from the S/T site, including 5387 positive samples and 5387 negative samples. A total of 204 samples were obtained from the Y site, including 102 positive samples and 102 negative samples. The above datasets were randomly divided into a non-overlapping training set and an independent test set with a ratio of 8:2 [18]. The datasets are showed in Table 1.

Table 1
The number of samples on training datasets and independent test datasets.

	Datasets	S/T	Y
Training set	Positive	4308	81
	Negative	4308	81
Independent test set	Positive	1079	21
	Negative	1079	21

2.2. Feature encoding algorithm

DDE: The deviation of dipeptide from expected mean (DDE) method was proposed by Saravanan et al. [19]. The DDE eigenvector was obtained by combining the dipeptide composition (DC), theoretical mean (TM) and theoretical variance (TV). Each protein sequence can obtain a 400-dimensional feature vector, and the formula is as follows:

$$DC(f,l)=\frac{N_{fl}}{n-1},f,l\in\{A,C,D,\cdots,Y\} \tag{1}$$

where N_{fl} is the number of amino acid pairs, fl and n are the length of the protein sequence.

$$TM(f,l)=\frac{C_f}{C_n}\times\frac{C_l}{C_n} \tag{2}$$

where C_f is the number of codons encoding the first amino acid, C_l is the number of codons encoding the second amino acid, and C_n is the total number of possible codons.

$$TV(f,l)=\frac{TM(f,l)(1-TM(f,l))}{n-1} \tag{3}$$

$$DDE(f,l)=\frac{DC(f,l)(1-TM(f,l))}{\sqrt{TV(f,l)}} \tag{4}$$

CTD: Composition, Transition and Distribution (CTD) [20–22] is used to characterize the physicochemical properties of antimicrobial peptide sequences and the distribution pattern of amino acid sequence composition. Taking hydrophobicity as an example, the replacement sequence consists of polar, neutral and hydrophobic residues of the protein, and the composition descriptor (C) is shown in formula (5):

$$C(r)=\frac{N(r)}{N},r\in\{polar,neutral,hydropholic\} \tag{5}$$

where $N(r)$ is the number of amino acid type r in the coding sequence and N is the sequence length.

The transition descriptor (T) includes three features, dipeptide composition frequency from polar to neutral and from neutral to polar, dipeptide composition frequency from neutral to hydrophobic and from hydrophobic to neutral, and dipeptide composition frequencies from polarity to hydrophobicity and from hydrophobicity to polarity. The T descriptor is defined as follows:

$$\begin{gathered}T(r,s)=\frac{N(r,s)+N(s,r)}{N-1}\\ r,s\in\{(polar,neutral),(neutral,hydrophobic),(hydrophobic,polar)\}\end{gathered} \tag{6}$$

where $N(r,s)$ and $N(s,r)$ are the dipeptide frequencies encoded as rs and sr, respectively, and N is the sequence length.

The distribution descriptor (D) consists of the location score for each of the three groups. For any given set of residues, the distribution descriptor is obtained by dividing the position of the residue by its length. The C, T and D descriptors generate 39, 39 and 195-dimensional feature vectors, respectively, and CTD generates a total of 273-dimensional feature vector. At present, the CTD algorithm has been extensively utilized in bioinformatics, such as the prediction of protein-protein interactions [23].

EAAC: Enhanced amino acid composition (EAAC) [24] is commonly used to encode protein sequences of equal length. The amino acid frequency of fixed-length subsequences is calculated by sliding consecutively from the N-terminus to the C-terminus of each protein sequence. The EAAC feature extraction algorithm finally obtains a 580-dimensional feature vector, and the algorithm is defined as follows:

$$g(r, win) = \frac{N(r, win)}{N(win)}, r \in (A, C, D, \cdots, Y) \tag{7}$$

among them, $N(r, win)$ is the number of amino acid types in the sliding window win, $win \in (window1, window2, \cdots, windowN)$, $N(win)$ is the size of the sliding window win, and the default value of the sequence window for $N(win)$ is 5.

AAindex: Amino acid property encodings (AAindex) is an amino acid index database [25], representing the physicochemical properties of amino acids and amino acid pairs. We take the physicochemical properties of 16 amino acids selected by Xiang et al. [26] from AAindex database as the research object. Each physicochemical property of amino acids is quantitatively scored and the mean value strategy is adopted. For each physicochemical characteristic, the corresponding fraction of each amino acid is added to the corresponding fraction of each amino acid in the window and divided it by the window size to obtain the physicochemical properties corresponding to the protein sequence, and 16 physicochemical property fraction is obtained for each amino acid residue. For the protein sequence with length L, a $16 * L$-dimension eigenvector is obtained.

PseAAC: Pseudo-amino acid composition (PseAAC) [27] extract the positional and compositional information of protein sequences. The feature vector of this feature extraction algorithm is expressed as:

$$L = [l_1, l_2, l_3, \cdots, l_{20}, l_{20+1}, \cdots, l_{20+\lambda}]^T \ , \ (\lambda < N) \tag{8}$$

where, the first 20-dimensional feature vector represents the amino acid composition information of the protein sequence, the λ-dimension represents the position information of the protein sequence, and N is the length of the amino acid sequence.

The vector element l_m is calculated as follows:

$$l_m = \begin{cases} \dfrac{q_m}{\sum\limits_{i=1}^{20} q_i + \omega \sum\limits_{e=1}^{\lambda} \theta_e}, (1 \leq m \leq 20) \\ \dfrac{\omega \theta_{m-20}}{\sum\limits_{i=1}^{20} q_i + \omega \sum\limits_{e=1}^{\lambda} \theta_e}, (20 + 1 \leq m \leq 20 + \lambda) \end{cases} \tag{9}$$

where q_i is the normalized frequency value, θ_e is the e-layer sequence correlation factor calculated according to the equation, and ω is the weighting factor. The dimension of the PseAAC feature vector is $20 + \ \lambda$. The PseAAC algorithm has been extensively utilized in bioinformatics, such as prediction of ubiquitination site [28], identification of subcellular localization [29], identification protein submitochondrial locations [30].

BLOSUM62: BLOSUM62 can extract evolutionary information of proteins [31] and is often used to encode protein sequences of equal length. The BLOSUM62 matrix is based on the comparison of amino acid sequences, and then we use the amino acid substitution matrix to measure the similarity of two amino acid segments, so that the identity of the two residue sequences does not exceed 62%. For a protein sequence of length L, a feature vector of $20 \times L$ dimension can be obtained by this algorithm.

2.3. Multi-view feature learning

Currently, most researchers simply fuse feature-extracted vectors in an end-to-end combinatorial manner. This simple end-to-end stitching ignores the relative weights of individual feature extraction methods and cannot capture more effective information comprehensively. Therefore, we adopt the DE algorithm [32] to solve the global optimization problem, and perform multi-information fusion through the given feature weights to achieve the best prediction effect. The DE algorithm has been applied by researchers in many fields. For example, researchers used DE algorithm to deal with neural network, bioinformatics and other related problems [33,34]. In this paper, DE algorithm mainly includes initialization, mutation, crossover and selection process. The algorithm steps are as follows.

2.3.1. Initialize population and parameters

Solve the goal optimization problem,

$$\begin{aligned} &\max f(w_1, w_2, \cdots w_D) \\ &s.t\ w_j^L \leq w_j \leq w_j^U, j = 1, 2, \cdots D \end{aligned} \tag{10}$$

among them, $f(w_i)$ is the maximum value representing the search AUC, D is the number of feature extraction methods, and in this paper $D = 6$, w_j^L and w_j^U represent the upper and lower bounds of the j-th component and $w_j \in [-\ 2, 2]$. The built-in parameter crossover probability CR is 0.5, the scaling factor F is 0.5, the population size N is 100, and the maximum number of iterations $G_{\max}$ is 100.

$P^g = \{w_1^g, w_2^g, \cdots w_N^g\}$ represents the random generation of the initial population, where N is the population size, and $w_i^g = (w_{i,1}^g, w_{i,2}^g, \cdots w_{i,6}^g)^T$ is the i-th individual of the g-th generation in the population.

2.3.2. Population variation

In order to complete the mutation between individual samples, the DE algorithm arbitrarily extracts different samples in P^g, and integrates the samples that need to be mutated with the scaled feature vector of the extracted samples, that is:

$$m_i^{g+1} = w_{r_1}^g + F \cdot \left(w_{r_2}^g - w_{r_3}^g \right) \tag{11}$$

where $w_{r_1}^g$, $w_{r_2}^g$ and $w_{r_3}^g$ are three randomly different individuals in the subset $P^g - \{w_i^g\}$.

2.3.3. Crossover of species

In order to obtain the g-generation population w_i^g and the sample m_i^{g+1} that have completed the previous step, this paper adopts crossover measures within multiple samples, so that the samples obtained in the next generation have multiple possibilities. Which is:

$$t_{i,k}^{g+1} = \begin{cases} m_{i,k}^{g+1}, \text{if } \ \text{rand}(0, 1) < CR \text{ or } k = k_{rand} \\ w_{i,k}^g, \text{otherwise} \end{cases} \tag{12}$$

where k_{rand} is a random integer with $[1, 2, \cdots D]$.

2.3.4. Choose the best population

The DE algorithm refers to using the function f to obtain the optimal sample by comparing the target sample w_i^g and the $t_{i,k}^{g+1}$ obtained in the previous step, namely:

$$w_i^{g+1} = \begin{cases} t_i^{g+1}, if\ f\left(t_i^{g+1}\right) \leq f(w_i^g) \\ w_i^g, \text{otherwise} \end{cases} \tag{13}$$

when $g > G_{\max}$, the algorithm is terminated and outputs w_{best}. Otherwise repeat steps (11) to (13). In this paper, we use the DE algorithm to obtain the feature weight values of six feature extraction algorithms, including PseAAC, CTD, DDE, BLOSUM62, AAindex and EAAC, as follows: $w_{best} = (0.1875, 0.6163, 0.2006, 1.2702, 0.5822, 1.5655)$.

2.4. Group least absolute shrinkage and selection operators

On the basis of the LASSO algorithm [35], Yuan et al. [36] proposed a group least absolute shrinkage and selection operators (Group LASSO) to further optimize the algorithm and better remove noise and redundancy. In the optimization process, the coefficient vectors are grouped by the algorithm. The above operations can improve the sparsity between feature groups. This method adds an penalty function to the coefficient vector based on the least squares method by grouping all variables, so that the coefficient compression with a small absolute value can be set to 0. Then, the feature groups with zero coefficients are discarded, and the feature groups with non-zero coefficients are retained. The algorithm is called Group LASSO and it is defined as follows:

$$min\left(\|Y - X\beta\|_2^2 + \lambda \sum_{r=1}^{R} \left\| \sqrt{ql}\beta I_r \right\|_2 \right) \tag{14}$$

where, Y represents N observation variables, P features are divided into groups, X is a $N \times P$ feature matrix, the value of i is $1, 2, \cdots, g, \cdots, G$, I_r is the feature subscript of the g group, and $\sqrt{ql}$ is the weight of each group.

2.5. Deep learning framework

The deep network architecture constructed in this paper is composed of a combination of multi-head attention mechanism [37–39], LSTM [40,41] and fully connected neural network layers, and it is called MULSTMFC. The deep network framework combines the advantages of multiple components to make learning more efficient. Next, all the components in the framework from input to output will be introduced in order.

2.5.1. Multi-head attention mechanism

In order to better capture the interaction information between protein sequences, this paper adopts the multi-head self-attention mechanism [37–39] in the framework of deep learning. The multi-head attention integrates information from different subspaces generated by multiple self-attention layers. The attention function is a linear mapping process of Q (query), K (key) and V (value), and the calculation result of the attention matrix is obtained by formula (15). The formula is described as follows:

$$Attention(Q, K, V) = \mathrm{softmax}\left(\frac{QK^T}{\sqrt{dk}}\right)V \tag{15}$$

where T represents the transpose operation of the matrix, d_k is a scale factor, and softmax is a determinant normalization function. For the multi-head attention module, first, the Q, K and V parameter matrices are obtained by performing i transformations through linear projection, and then the obtained Q, K and V parameter matrices are input into g parallel heads for attention matrix operations respectively. Through the above steps, each parallel head can obtain feature information from different self-attention subspaces. Finally, the output result ST of the g parallel heads is obtained through the connection operation. The specific process can be described by formulas (16) and (17):

$$head_i = Attention\left(QW_i^Q, KW_i^k, VW_i^V\right) \tag{16}$$

$$ST_t = MutiHhead = (Q, K, V) = Concat\left(head_1 \cdots, head_g\right)W^D \tag{17}$$

where W_i^Q, W_i^k, W_i^V, and W^D are the parameter matrices used for the corresponding linear projections. $Head_r$ represents the g-th head, in this paper $g = 5$, and Concat represents the concatenation operation.

2.5.2. Long short-term memory network

LSTM [40,41] is mainly used to solve the long-term dependency problem in the recurrent neural network, and it is a special recurrent neural network. However, in addition to the internal processing units, it differs in the use of the gates of the recurrent neurons of the forget gate, update gate and output gate. The division of labor responsible for each door is different. The calculation of the forget gate is expressed as:

$$f_l = \sigma\left(W_f h_{l-1} + U_f x_l + b_f\right) \tag{18}$$

The input gate has two parts. The first is that a sigmoid layer can decide which values need to be updated. Next, there is a tanh layer which contains the newly added information. The unit state is updated by combining the information of the two parts. The calculation of the two parts is expressed as follows:

$$i_l = \sigma(W_r h_{l-1} + U_r x_l + b_r) \tag{19}$$

$$a_l = tanh(W_a h_{l-1} + U_a x_l + b_a) \tag{20}$$

where W_r, U_r, W_a and U_a are linear correlation coefficients, b_r and b_a are the bias term of the forget gate, and h_{l-1} indicates the concealment of the previous layer at time $t - 1$ or the original concealment at time 0.

The output results of the input gate and forget gate are combined for the cell state C_l, which comprises of two parts, and its formula can be showed as:

$$C_l = C_{l-1} \otimes f_l + i_l \otimes a_l \tag{21}$$

Next, we introduce the output gate. The sigmoid function is used to calculate what information needs to be output, the output is h_l, and then it is multiplied by the current unit state through the value of the activation function tanh as follows:

$$o_l = \sigma(W_o h_{l-1} + U_0 x_l + b_o) \tag{22}$$

$$h_l = o_l \otimes \tanh(C_l) \tag{23}$$

Finally, we input the z_l obtained in the previous operation step into the FCN network layer, and use the following formula to output the result.

$$Y_t = z_l W_{hq} + b_q \tag{24}$$

Here, the model in this paper uses Dropout to prevent overfitting of the model, and selects ReLU as the activation function of the model. Adam function is used as an optimizer to update model parameters, and categorical_crossentropy is used as loss function to measure the accuracy of model classification. In the binary classification problem, we use softmax function to classify SARS-CoV-2 phosphorylation sites.

2.6. Performance evaluation

To evaluate the effectiveness of our deep learning model, we adopt accuracy (ACC), specificity (SPEC), sensitivity (SEN) and MCC [42–44] as model performance evaluation metrics. The above indicators are described as follows:

$$SEN = \frac{TP}{TP + FN} \tag{25}$$

$$SPEC = \frac{TN}{TN + FP} \tag{26}$$

$$ACC = \frac{TP + TN}{TP + FP + TN + FN} \tag{27}$$

$$MCC = \frac{TP \times TN - FP \times FN}{\sqrt{(TP + FP) \times (TP + FN) \times (TN + FP) \times (TN + FN)}} \tag{28}$$

among them, TP represents the number of correctly predicted SARS-CoV-2 phosphorylation sites, and TN represents the number of correctly identified non-SARS-CoV-2 phosphorylation sites. FP represents the number of erroneously identified phosphorylation sites for SARS-CoV-2, and FN represents the number of erroneously identified non-SARS-CoV-2. And, ROC and PR curves are also important metrics for evaluating classification accuracy. The AUC and AUPR values [45,46] represent the area under the ROC curve and the PR curve, respectively.

Statistical test methods, two-tailed T-test [47] is often used to evaluate the predictive performance of the model. The formula is expressed as follows:

$$t=\frac{\overline{Q_1}-\overline{Q_2}}{\sqrt{\frac{(p_1-1)R_1^2+(p_2-1)R_2^2}{p_1+p_2-2}\left(\frac{1}{p_1}+\frac{1}{p_2}\right)}} \tag{29}$$

where $\overline{Q_i}, R_i^2$ represent the mean value, sample variance, respectively, and p_i capacitance of the i-th sample date, respectively.

For convenience, the method proposed in this study to identify SARS-CoV-2 phosphorylation site is called DE-MHAIPs, and its workflow can be seen in Fig. 1. The experimental environment of our model is: Python 3.7 and R ×64 3.6.2 programming, with 32.0 GB of RAM. The source codes and datasets are available at https://github.com/YuBinLab-QUST/DE-MHAIPs/.

The DE-MHAIPs model can be described as the following steps:

The workflow of the DE-MHAIPs model describes the steps as follows:

Step1. data acquisition. Experimentally validated S/T and Y data are collected from the literature. The data include protein sequences and corresponding labels.

Step2. Feature extraction. Six feature extraction methods (AAindex, CTD, DDE, EAAC, PseAAC, and BLOSUM62) are used to obtain 528, 273, 400, 580, $20+\lambda$ and 660-dimensional feature vectors, respectively. The DE optimization algorithm is used to perform multi-view weighted fusion of the six features.

Step3. Feature selection. We use Group LASSO to reduce the dimensionality of high-dimensional data obtained by multi-view feature fusion, so as to obtain excellent feature subsets.

Step4. Construction of deep learning network framework. The optimal feature subsets obtained by feature selection are separately input into the MULSTMFC classifier for predicting the SARS-CoV-2 phosphorylation sites.

Step5. Model performance evaluation. We use 5-fold cross-validation to evaluate the model constructed in Step1-Step4. The AUC, AAC, SEN, SPEC, and MCC metric values are used to evaluate the results of the prediction system. In addition, AUC and AUORP values are also used to evaluate the predictive ability of the model.

Step6. According to the model constructed in Step1-Step5, independent test sets is used to test the model.

3. Results and discussion

3.1. Sequence analysis

In order to more intuitively distinguish the distribution of amino acid residues at SARS-CoV-2 phosphorylation sites and non-SARS-CoV-2 phosphorylation sites, we use Two-Sample-Logo [48] to draw SARS-CoV-2 phosphorylation sites Sequence map of fragments surrounding the phosphorylation site and non-SARS-CoV-2 phosphorylation site. The sequence length of the dataset is 33, so the dataset is described in the range of −16 to 16. The sequence analysis of positive and negative samples is shown in Supplementary Fig. S1.

3.2. Influence of feature extraction algorithm parameters on results

The PseAAC feature extraction method needs to determine the optimal parameters when performing feature extraction on proteins. The parameter setting range of PseAAC is 1–32, and the interval is 1. The

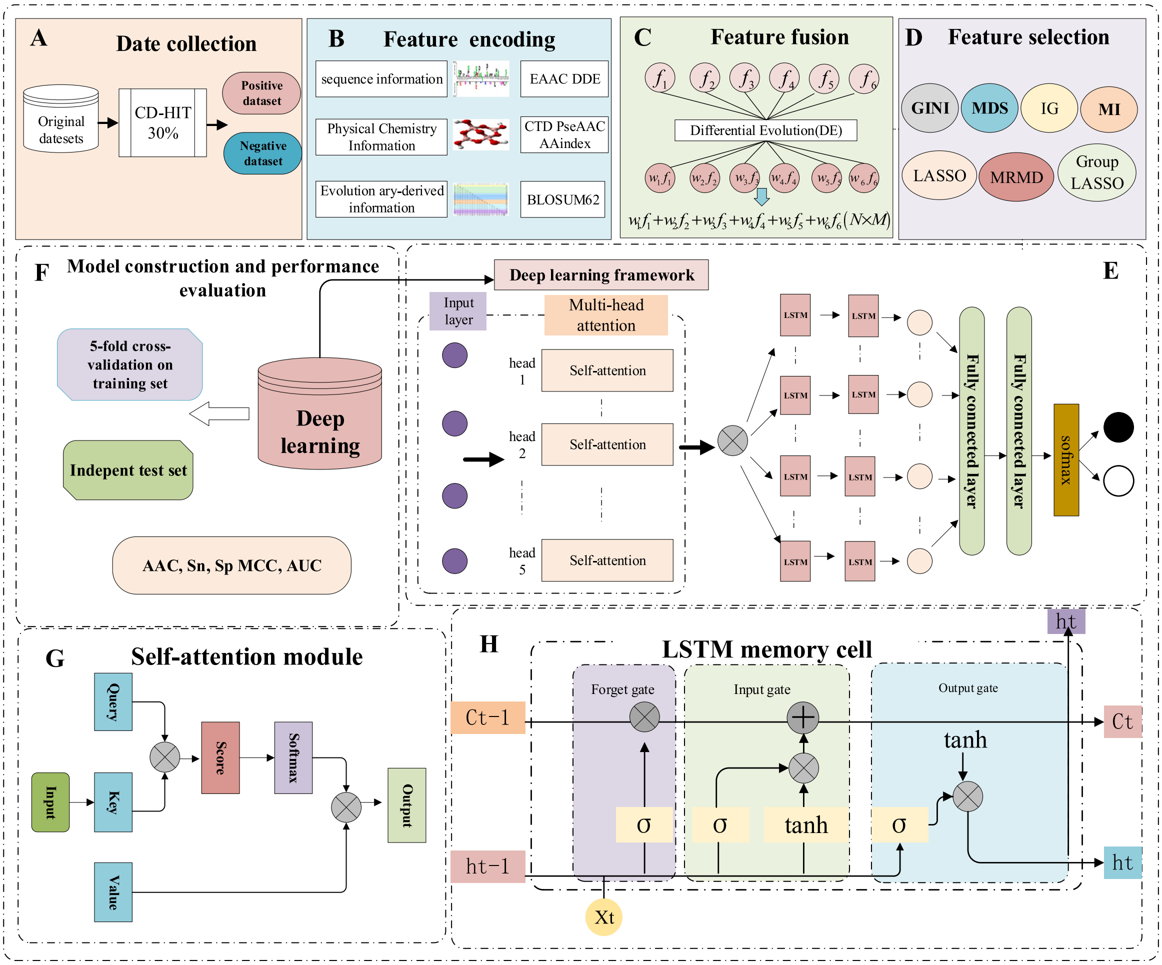

Fig. 1. Schematic diagram of the DE-MHAIPs prediction method.

ACC and MCC are used as evaluation indicators to select the best feature parameters. The prediction results corresponding to different parameters of the S/T and Y datasets on the training set are shown in Supplementary Fig. S2. The predicted values of specific indicators can be seen in Supplementary Tables S1–S2.

3.3. Influence of feature extraction algorithms on training sets on results

In this paper, a variety of protein information is comprehensively considered, and the sequence information of protein is extracted by six methods: CTD, DDE, EAAC, PSEAAC, AAindex and BLOSUM62. At the same time, we use two multi-information fusion methods to fuse these six feature extraction methods. One is to simply combine the six feature extraction methods in an end-to-end manner, namely: AAindex + CTD + DDE + EAAC + PseAAC + BLOSUM62. The other is to learn the weight of the single feature extraction method according to the DE algorithm, so as to perform multi-information fusion according to the single feature weight, namely: AAindex w_1 + CTD w_2 + DDE w_3 + EAAC w_4 + PseAAC w_5 + BLOSUM62 w_6, We use the DE algorithm to obtain the feature weight values of six feature extraction algorithms, including, PseAAC, CTD, DDE, BLOSUM62, AAindex and EAAC. And we obtain the feature weight values of the six feature extraction algorithms, as follows: $w_{best} = (0.1875, 0.6163, 0.2006, 1.2702, 0.5822, 1.5655)$. A 2487-dimensional feature vector is obtained on the S/T data set, and a 2492-dimensional feature vector is obtained on the Y data set. The deep learning framework MULSTMFC is used as a classifier to predict the phosphorylation sites of SARS-CoV-2. The ROC and PR curves obtained from the S/T and Y datasets are shown in Fig. 2, and the predicted values of each index obtained by a single feature and the fusion of the two methods can be seen in Supplementary Table S3.

As shown in Fig. 2, on the S/T and Y datasets, the AUC values obtained by end-to-end fusion are 88.26% and 80.71%, respectively, which are 0.88%–7.93% and 1.28%–13.42% higher than AAindex, CTD, DDE, EAAC, PseAAC and BLOSUM62 single feature methods. The AUC values obtained by using the DE optimization algorithm for multi-information fusion are 89.76% and 82.06%, which are 1.50% and 1.35% higher than the AUC values obtained by end-to-end fusion, respectively. By analyzing the index prediction results, the DE algorithm multi-feature fusion algorithm is used to achieve better results and has better generalization ability.

3.4. The effect of feature selection on prediction results

In binary classification problems, in order to better obtain the optimal feature subset, this paper selects six dimensionality reduction methods for feature selection, including GINI [49], IG [50], MDS [51], MI, LASSO [52], MRMD [53]. For the sake of fairness, the parameters of LASSO and Group LASSO are both set to 0.03, and the optimal feature subsets of 373 and 347 dimensions are obtained respectively. The optimal number of features retained by GINI, MI, IG, MDS, and MRMD is consistent with that of Group LASSO. The feature samples obtained by different dimensionality reduction methods are input into the MULSTMF classifier. The prediction results of different indicators under 5-fold cross-validation are shown in Table 2. In addition, we have also plotted a boxplot of six dimensionality reduction methods on AUC values, which are shown in Fig. 4. The specific AUC values are shown in Supplementary Table S4.

As can be seen from Table 2, on the S/T data set, the ACC, MCC and AUC values of the Group LASSO feature selection reached 84.42%, 68.48% and 91.98%, respectively, which exceeded the other five feature selection methods. They are 2.15%–4.62%, 3.00%–8.81%, and 1.01%–4.04% higher than other methods, respectively. Among them, the results of each indicator of MDS are lower and the performance is the worst. Similarly, on the Y dataset, the ACC, MCC, and AUC values after Group LASSO feature selection are 93.24%, 86.67%, and 98.32%, respectively. The results of each index are higher than other feature selection methods. From Fig. 3, the group LASSO method achieves the highest AUC value on the S/T and Y datasets. It shows that Group LASSO can

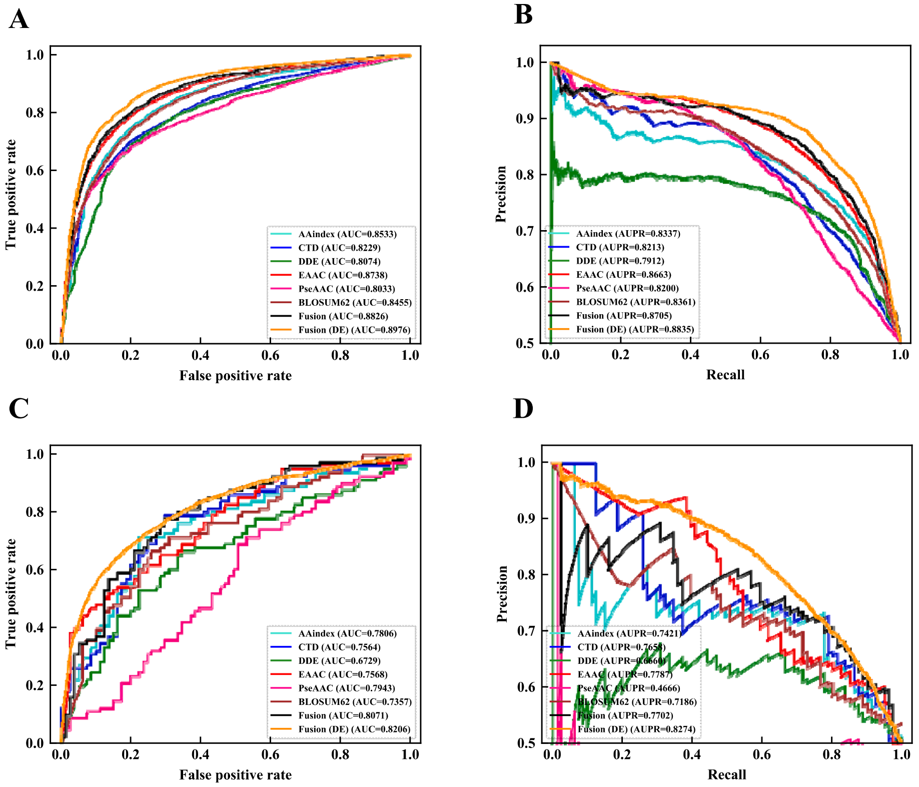

Fig. 2. ROC curves of six feature extraction methods and two multi-information fusion methods. (A–B) ROC and PR curves of different feature extraction methods on the S/T dataset. (C–D) ROC and PR curves of different feature methods on the Y dataset.

Table 2
Influence of feature selection methods on predicted values.

Datasets	Methods	ACC (%)	SEN (%)	SPEC (%)	MCC	AUC (%)
S/T	GINI	81.03	81.19	80.87	0.6211	88.92
	IG	81.62	84.49	78.76	0.6340	89.85
	MDS	80.80	82.42	79.17	0.6168	89.25
	MI	81.24	83.70	78.78	0.6257	89.28
	MRMD	79.80	78.80	80.79	0.5967	87.94
	LASSO	82.27	83.56	81.89	0.6548	90.97
	Group LASSO	**84.42**	**84.58**	**83.89**	**0.6848**	**91.98**
Y	GINI	85.73	83.89	87.64	0.7164	91.22
	IG	79.07	84.04	74.04	0.5860	87.47
	MDS	74.05	70.22	78.01	0.4964	82.90
	MI	78.97	82.79	75.29	0.5866	83.20
	MRMD	82.68	83.89	81.54	0.6554	91.97
	LASSO	91.96	93.75	90.07	0.8414	98.11
	Group LASSO	**93.24**	**95.07**	**91.32**	**0.8667**	**98.32**

better filter out the noise caused by multi-feature fusion and can filter out more effective feature information for the classifier.

3.5. Feature visualization before and after comparison

In this paper, we use t-SNE [54,55] to visualize protein data information before and after the Group LASSO process. The t-SNE is mainly to better project data points with high similarity in multidimensional space into low-dimensional space, and to more intuitively show the comparison between positive and negative sample data before and after. The model training results are plotted separately in 2D coordinates, and the visualization results are displayed in Supplementary Fig. S3.

3.6. The effect of different classification models on the results

This paper selects 9 classifier algorithms for comparison, including AdaBoost [56], NB [57], RF [58], XGBoost [59], CNN [60], LSTM [61], CNN-LSTM, LSTM-DNN [62,63], MULSTMFC, where AdaBoost, NB, Bagging are the default parameters for the three classifiers, and the XGBoost learning rate is set to 0.01. The CNN consists of two convolutional layers and a max pooling layer, followed by a FCN layer for activation using the ReLU function. The LSTM consists of two LSTM layers and a FCN layer each using a sigmoid as the activation function. CNN-LSTM consists of CNN using two convolutional layers and one max pooling layer followed by two LSTM layers and one FCN layer. LSTM-DNN consists of two LSTM layers and three FCN layers. In this paper MULSTMFC includes a multi-head attention layer, two LSTM layers and two FCN layers.

The vectors after feature selection are input into different classification models and deep learning frameworks. The results under 5-fold cross-validation on the training set are shown in Table 3. We also plotted ROC and PR curves to more intuitively compare the performance of the nine classifiers, and the results are shown in Fig. 4. In addition, in order to further verify the performance of the MULSTMFC classifier, we performed a two-sided T-test on the evaluation indicators of different classifiers to compare whether there is a significant difference between the indicators results of the other classifiers. At significance level $\alpha = 0.05$, we compared AdaBoost, NB, RF, XGBoost, CNN, LSTM, CNN-LSTM and LSTM-DNN with MULSTMFC classifier P-values on ACC and MCC metrics. The statistical test results of the bilateral T-test can be seen in Table 4.

As can be seen from Table 3, on the S/T dataset, the ACC values corresponding to the nine classifiers are 81.27%, 76.84%, 81.98%, 81.84%, 80.26%, 81.83%, 80.29%, 81.56%, and 84.23%, respectively. The deep learning model MULSTMFC constructed in this paper shows the best prediction effect. The ACC and AUC values are 84.23% and 91.98%, which are 2.25%–7.39% and 2.07%–7.53% higher than other models, respectively. Similarly, on the Y dataset, the deep learning model MULSTMFC constructed in this paper shows the best prediction effect compared with the other 8 classification models, with ACC and AUC values of 93.24% and 98.32%, which are 1.28%–10.57% and 0.06%–10.46% higher than other models. Therefore, in this study, MULSTMFC is used as the best classification model for predicting phosphorylation sites of SARS-CoV-2.

As can be seen from Fig. 4, on the S/T and Y datasets, the ROC and PR curves of the MULSTMFC classifier clearly cover other classification models. On the S/T dataset, the ROC and PR curve values are 91.98% and 92.12%, which are 0.82%–7.05% and 0.26%–8.68% higher than other classifier, respectively. On the Y dataset, the ROC and PR curve values are 98.32% and 98.39%, which are 0.05%–10.45% and 1.35%–14.72% higher than other classifiers, respectively. The ROC and PR curve values on both datasets are higher than the other 8 classifiers, achieving the best prediction effect. Therefore, the MULSTMFC classifier shows better robustness and generalization ability, and can more accurately predict the phosphorylation sites of the SARS-CoV-2.

In order to comprehensively evaluate the classification ability of MULSTMFC, we introduce a two-sided T-test [47], that is, the T distribution theory is used to infer the probability of difference occurrence. Based on the 5-fold cross-validation, the T-test can be used to compare whether the difference between the values of different classifier evaluation indicators is significant, so as to evaluate the predictive ability of the classifier. The T-test results of different classifiers are shown in Table 4.

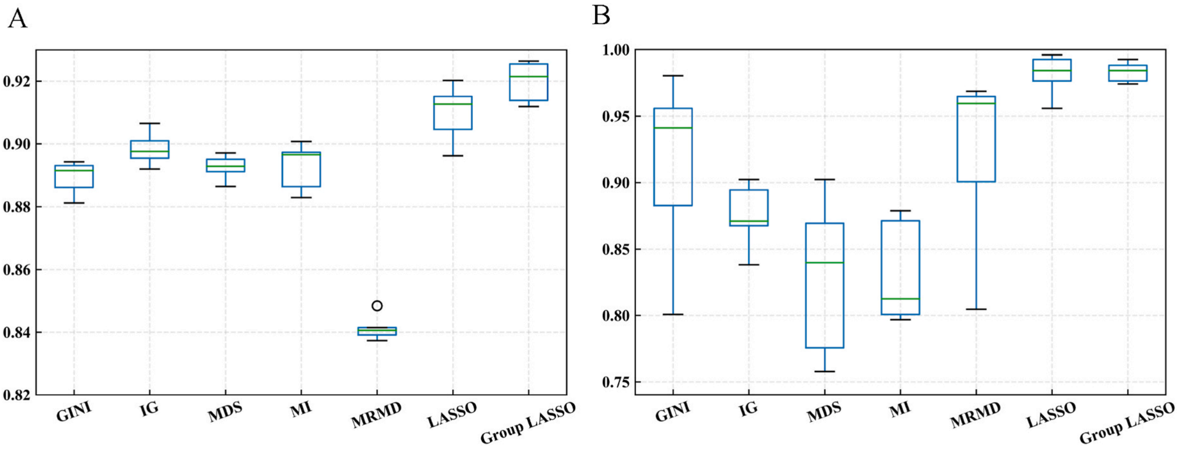

Fig. 3. Boxplot of AUC values corresponding to seven dimensionality reduction methods. (A) Boxplots of different methods on the S/T dataset. (B) Boxplots of different methods on the Y dataset.

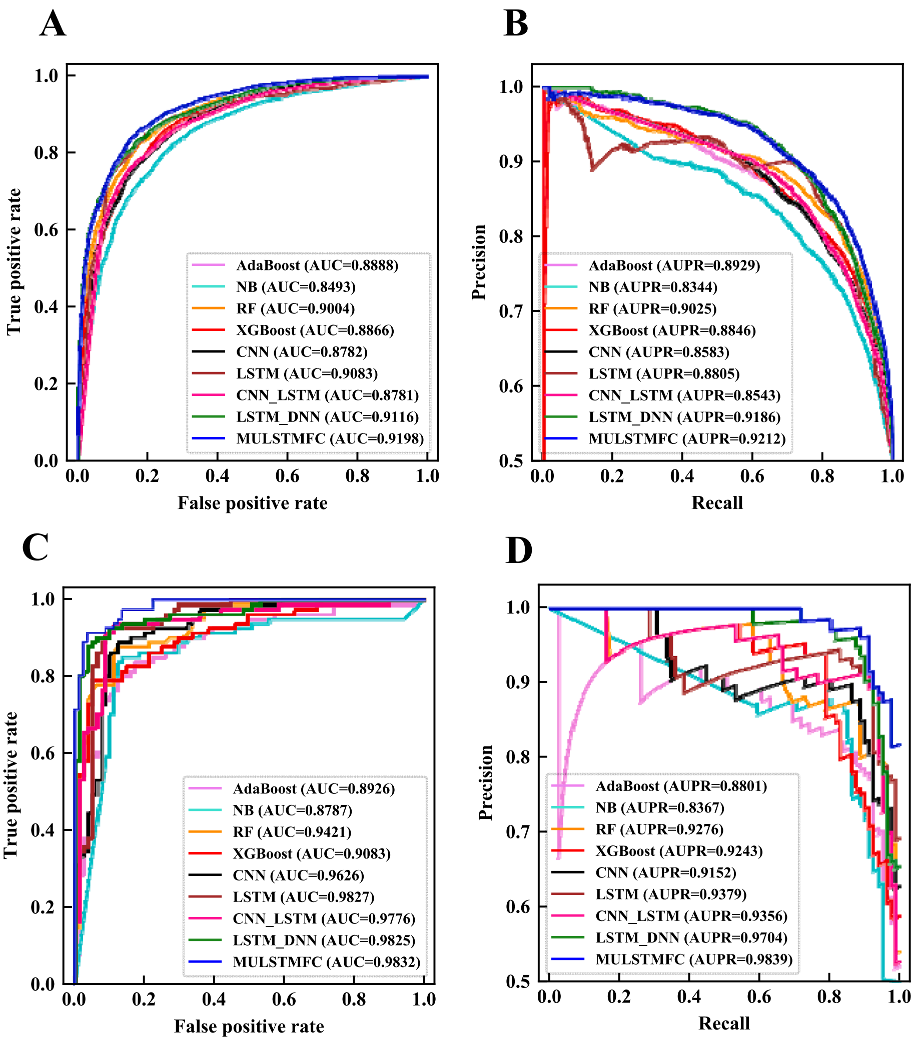

Fig. 4. The ROC and PR curves of AdaBoost, NB, RF, XGBoost, CNN, LSTM, CNN_LSTM, LSTM_DNN and MULSTMFC classifiers on training datasets. (A–B) ROC and PR curves of different classification models on the S/T dataset. (C–D) ROC and PR curves on the Y dataset for different classification models.

Table 3
The effect of different classification models on the results.

Datasets	Classifier	ACC (%)	SEN (%)	SPEC (%)	MCC	AUC (%)
S/T	AdaBoost	81.27	82.24	80.31	0.6257	89.01
	NB	76.84	83.28	70.40	0.5415	84.45
	RF	81.98	81.52	82.47	0.6403	89.91
	XGBoost	81.84	82.31	81.38	0.6371	89.28
	CNN	80.26	80.13	80.40	0.6057	87.79
	LSTM	81.83	82.56	78.22	0.6425	89.81
	CNN-LSTM	80.29	80.43	80.15	0.6063	87.68
	LSTM-DNN	81.56	82.74	78.38	0.6331	89.69
	MULSTMFC	**84.23**	**84.58**	**83.89**	**0.6848**	**91.98**
Y	AdaBoost	82.67	87.50	77.79	0.6640	89.26
	NB	85.81	86.32	85.14	0.7201	87.86
	RF	86.43	88.97	84.12	0.7405	94.21
	XGBoost	83.88	85.00	82.72	0.6812	90.83
	CNN	86.95	84.97	88.23	0.7521	96.25
	LSTM	90.09	93.75	86.39	0.8162	98.26
	CNN-LSTM	90.72	87.79	91.27	0.8298	97.76
	LSTM-DNN	91.96	93.82	90.00	0.8425	98.24
	MULSTMFC	**93.24**	**95.07**	**91.33**	**0.8667**	**98.32**

From Table 4, in terms of ACC and AUC, at the significance level $\alpha = 0.05$, the MULSTMFC classifier mostly outperforms AdaBoost, NB, RF, XGBoost, CNN, LSTM, CNN-LSTM and LSTM-DNN classification models.

Table 4
P value of MULSTMFC with other classifiers on ACC and AUC indicator.

Classifier	ACC		AUC	
	S/T	Y	S/T	Y
AdaBoost	3.02E-05	5.73E-03	2.98E-05	6.63E-03
NB	1.19E-06	2.74E-02	1.58E-06	3.03E-02
RF	5.72E-03	2.83E-02	5.38E-03	3.64E-02
XGBoost	6.54E-04	5.55E-02	6.56E-04	5.59E-02
CNN	6.91E-04	1.51E-01	6.89E-04	1.56E-01
LSTM	3.79E-02	1.85E-01	3.83E-02	2.55E-01
CNN-LSTM	1.36E-04	3.99E-01	1.32E-04	4.92E-01
LSTM-DNN	2.23E-02	6.56E-01	2.43E-02	6.66E-01

The results show that the MULSTMFC classifier has significant differences compared with other classifiers, and show a relatively stable statistical significance. So, we adopted the MULSTMFC classification model as the best classifier to identify phosphorylation sites of SARS-CoV-2.

3.7. Comparison with others' results on an independent test set

To further evaluate the effectiveness of the DE-MHAIPs model, we use an independent test set to compare the evaluation DE-MHAIPs with several existing phosphorylation site prediction models, DeepIPs [15], DeepPSP [14] MusiteDeep2020 [64], MusiteDeep2017 [12], the results

Table 5
Comparison with the results of others.

Residue type	Method	ACC (%)	SEN (%)	SPEC (%)	MCC	AUC (%)
S/T	DE-MHAIPs	**83.71**	**84.28**	83.14	**0.6745**	**91.72**
	DeepIPs	80.63	79.61	83.50	0.6316	89.37
	DeepPSP	80.21	76.65	**83.78**	0.6058	87.62
	MusiteDeep2020	80.95	82.95	78.96	0.6196	88.67
	MusiteDeep2017	80.17	78.87	81.46	0.6035	87.98
Y	DE-MHAIPs	**91.40**	95.07	**87.86**	**0.8375**	**97.78**
	DeepIPs	83.33	90.48	80.95	0.7175	92.52
	DeepPSP	76.19	**95.24**	57.14	0.5665	82.09
	MusiteDeep2020	85.51	95.24	76.19	0.7276	87.30
	MusiteDeep2017	80.95	85.71	76.19	0.62.19	81.41

are shown in Table 5. In order to show the results of different prediction models more intuitively, a histogram is drawn in this paper, as shown in Supplementary Fig. S4.

From Table 5, the ACC, MCC, and AUC of the DE-MHAIPs prediction model proposed in this paper on the S/T dataset are 83.71%, 67.45%, 91.72%, respectively. The AUC is 2.35%, 4.10%, 3.05%, 3.74% higher than the DeepIPs, DeepPSP, MusiteDeep2020 and MusiteDeep2017 prediction models, respectively. The ACC, MCC, and AUC on the Y dataset are 91.40%, 83.75%, and 97.78%, respectively, and the AUC is 5.26%, 15.69%, 10.48%, and 16.37% higher than the DeepIPs, DeepPSP, MusiteDeep2020 and MusiteDeep2017 prediction models, respectively. The results of different indicators show that the DE-MHAIPs prediction model significantly improves the prediction accuracy and effectively improves the prediction performance of the model.

4. Conclusion

The identification of phosphorylation sites can further speculate on the host kinases involved, which will provide potential avenues for the pharmacological treatment of SARS-CoV-2. Computational models used to identify phosphorylation sites could accelerate the development of these novel drug candidates. In this paper, we propose a new identification model for phosphorylation of novel coronavirus, DE-MHAIPs. For the first time, the combination of multi-head attention mechanism and LSTM network is used to construct a deep learning framework to identify phosphorylation sites of SARS-CoV-2. First, protein sequences are converted into digital information using six feature extraction methods including sequence information, physicochemical properties and evolutionary information. By analyzing the different characteristics, EAAC, CTD and BLOSUM62 have higher contribution rates to the model. Among them, EAAC extracts the sequence information of proteins, CTD extracts the physicochemical information of proteins, and BLOSUM62 extracts the evolution-derived information of proteins. Next, the differential evolution algorithm is used to fuse multi-feature information, so as to improve the global search ability of the model under the condition of ensuring diversity. Then, the LASSO group is used to obtain the optimal feature vector. Traditional classification algorithms are weak in dealing with overfitting and vanishing gradients, the MULSTMFC classification model combines the advantages of multiple attention mechanism and LSTM two deep learning methods. The Multi-head attention mechanism can selectively assign higher weights to some important SARS-CoV-2 phosphorylation site information, which is conducive to capture richer feature information. The LSTM layer decides to retain or clear information according to the openness of the memory gate, forget gate and output gate, which facilitates error backpropagation and improves the predictive ability of the model. Finally, the FCN network layer is used to classify SARS-CoV-2 phosphorylation sites, and the FCN network can effectively avoid the problem of gradient disappearance. The experimental results show that the DE-MHAIPs method can effectively identify the phosphorylation sites of SARS-CoV-2 and the prediction performance is significantly improved.

Although the prediction model we constructed has significantly improved the prediction accuracy of identifying SARS-CoV-2 phosphorylation, there are still some flaws that need to be improved in the future. First of all, we need to organize and construct large-scale and high-quality SARS-CoV-2 phosphorylation sites datasets in response to updates and changes in high-throughput data. Secondly, we need to consider more effective optimization algorithms, perform combinatorial optimization, and improve the predictive ability of the model. And it is hoped that it can provide more meaningful guidance for the study of SARS-CoV-2 phosphorylation sites. In the future, we will combine some effective algorithms, such as graph attention neural network and generative adversarial neural network, to further improve the prediction performance of DE-MHAIPs.

Declaration of competing interest

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

Acknowledgments

This work was supported by the National Natural Science Foundation of China (No. 62172248), and the Natural Science Foundation of Shandong Province of China (No. ZR2021MF098).

Appendix A. Supplementary data

Supplementary data to this article can be found online at https://doi.org/10.1016/j.compbiomed.2023.106935.

References

[1] B. Hu, H. Guo, P. Zhou, Z. L Shi, Characteristics of SARS-CoV-2 and COVID-19, Nat. Rev. Microbiol. 19 (3) (2021) 141–154.

[2] M. Cao, X. Su, S. Jiang, Broad-spectrum anti-coronavirus vaccines and therapeutics to combat the current COVID-19 pandemic and future coronavirus disease outbreaks, Stem Cell Rep. 16 (3) (2021) 398–411.

[3] S.A. More, A.S. Patil, N.S. Sakle, S.N. Mokale, Network analysis and molecular mapping for SARS-CoV-2 to reveal drug targets and repurposing of clinically developed drugs, Virology 555 (2020) 10–18.

[4] D. Bojkova, K. Klann, B. Koch, M. Widera, D. Krause, S. Ciesek, J. Cinatl, C. Münch, Proteomics of SARS-CoV-2-infected host cells reveals therapy targets, Nature 583 (7816) (2020) 469–472.

[5] R.M. Hekman, A.J. Hume, R.K. Goel, K.M. Abo, J. Huang, B.C. Blum, R.B. Werder, E.L. Suder, I. Paul, S. Phanse, A. Youssef, K.D. Alysandratos, D. Padhorny, S. Ojha, A. Mora-Martin, D. Kretov, P.E.A. Ash, M. Verma, J. Zhao, J.J. Patten, C. Villacorta-Martin, D. Bolzan, C. Perea-Resa, E. Bullitt, A. Hinds, A. Tilston-Lunel, X. Varelas, S. Farhangmehr, U. Braunschweig, J.H. Kwan, M. McComb, A. Basu, M. Saeed, V. Perissi, E.J. Burks, M.D. Layne, J.H. Connor, R. Davey, J.X. Cheng, B.L. Wolozin, B.J. Blencowe, S. Wuchty, S.M. Lyons, D. Kozakov, D. Cifuentes, M. Blower, D. N. Kotton, A.A. Wilson, E. Mühlberger, A. Emili, Actionable cytopathogenic host responses of human alveolar type 2 cells to SARS-CoV-2, Mol. Cell. 80 (6) (2020) 1104–1122.

[6] M. Bouhaddou, D. Memon, B. Meyer, K.M. White, V.V. Rezelj, M.M. Correa, B. J. Polacco, J.E. Melnyk, S. Ulferts, R.M. Kaake, J. Batra, A.L. Richards, E. Stevenson, D.E. Gordon, A. Rojc, K. Obernier, J.M. Fabius, M. Soucheray, L. Miorin, E. Moreno, C. Koh, Q.D. Tran, A. Hardy, R. Robinot, T. Vallet, B. E. Nilsson-Payant, C. Hernandez-Armenta, A. Dunham, S. Weigang, J. Knerr, M. Modak, D. Quintero, Y. Zhou, A. Dugourd, A. Valdeolivas, T. Patil, Q. Li, R. Hüttenhain, M. Cakir, M. Muralidharan, M. Kim, G. Jang, B. Tutuncuoglu, J. Hiatt, J.Z. Guo, J. Xu, S. Bouhaddou, C.J.P. Mathy, A. Gaulton, E.J. Manners, E. Félix, Y. Shi, M. Goff, J.K. Lim, T. McBride, M.C. O'Neal, Y. Cai, J.C.J. Chang, D. J. Broadhurst, S. Klippsten, E. Wit, A.R. Leach, T. Kortemme, B. Shoichet, M. Ott, J. Saez-Rodriguez, B.R. tenOever, R.D. Mullins, E.R. Fischer, G. Kochs, R. Grosse, A. García-Sastre, M. Vignuzzi, J.R. Johnson, K.M. Shokat, D.L. Swaney, P. Beltrao, N.J. Krogan, The global phosphorylation landscape of SARS-CoV-2 infection, Cell 182 (2020) 685–712.

[7] K. Klann, D. Bojkova, G. Tascher, S. Ciesek, C. Münch, J. Cinatl, Growth factor receptor signaling inhibition prevents SARS-CoV-2 replication, Mol. Cell. 80 (2020) 164–174.

[8] H.Y.L. Tung, P. Limtung, Mutations in the phosphorylation sites of SARS-CoV-2 encoded nucleocapsid protein and structure model of sequestration by protein 14-3-3, Biochem. Biophys. Res. Commun. 532 (1) (2020) 134–138.

[9] Y. Dou, B. Yao, C. Zhang, PhosphoSVM: prediction of phosphorylation sites by integrating various protein sequence attributes with a support vector machine, Amino Acids 46 (6) (2014) 1459–1469.

[10] F. Li, C. Li, T.T. Marquez-Lago, A. Leier, T. Akutsu, A.W. Purcell, A.I. Smith, T. Lithgow, R.J. Daly, J. Song, K.C. Chou, Quokka: a comprehensive tool for rapid and accurate prediction of kinase family-specific phosphorylation sites in the human proteome, Bioinformatics 34 (24) (2018) 4223–4231.

[11] T.H. Dang, Q.T. Trac, H.K. Phan, M.C. Nguyen, Q.T.P. Thi, SKIPHOS, Non-kinase Specific Phosphorylation Site Prediction with Random Forests and Amino Acid Skip-Gram Embeddings, bioRxiv, 2019, 793794.

[12] D. Wang, S. Zeng, C. Xu, W. Qiu, Y. Liang, T. Joshi, D. Xu, MusiteDeep: a deep-learning framework for general and kinase-specific phosphorylation site prediction, Bioinformatics 33 (24) (2017) 3909–3916.

[13] S. Ahmed, M. Kabir, M. Arif, Z.U. Khan, D.J. Yu, DeepPPSite: a deep learning-based model for analysis and prediction of phosphorylation sites using efficient sequence information, Anal. Biochem. 612 (2021), 113955.

[14] L. Guo, Y. Wang, X. Xu, K.K. Cheng, Y. Long, J. Xu, S. Li, J. Dong, DeepPSP, A global-local information-based deep neural network for the prediction of protein phosphorylation sites, J. Proteome Res. 20 (1) (2021) 346–356.

[15] H. Lv, F.Y. Dao, H. Zulfiqar, H. Lin, DeepIPs: comprehensive assessment and computational identification of phosphorylation sites of SARS-CoV-2 infection using a deep learning-based approach, Briefings Bioinf. 22 (6) (2021) bbab244.

[16] A. Stukalov, V. Girault, V. Grass, O. Karayel, V. Bergant, C. Urban, D.A. Haas, Y. Huang, L. Oubraham, A. Wang, M.S. Hamad, A. Piras, F.M. Hansen, M.C. Tanzer, I. Paron, L. Zinzula, T. Engleitner, M. Reinecke, T.M. Lavacca, R. Ehmann, R. Wölfel, J. Jores, B. Kuster, U. Protzer, R. Rad, J. Ziebuhr, V. Thiel, P. Scaturro, M. Mann, A. Pichlmair, Multilevel proteomics reveals host perturbations by SARS-CoV-2 and SARS-CoV, Nature 594 (7862) (2021) 246–252.

[17] W. Li, A. Godzik, Cd-hit, A fast program for clustering and comparing large sets of protein or nucleotide sequences, Bioinformatics 22 (2006) 1658–1659.

[18] F. Luo, M. Wang, Y. Liu, X.M. Zhao, A. Li, DeepPhos: prediction of protein phosphorylation sites with deep learning, Bioinformatics 35 (2019) 2766–2773.

[19] Z. Chen, P. Zhao, F. Li, A. Leier, T.T. Marquez-Lago, Y. Wang, G.I. Webb, A.I. Smith, R.J. Daly, K.C. Chou, J. Song, iFeature: a python package and web server for features extraction and selection from protein and peptide sequences, Bioinformatics 34 (2018) 2499–2502.

[20] C.Z. Cai, L.Y. Han, Z.L. Ji, X. Chen, Y.Z. Chen, SVM-Prot: web-based support vector machine software for functional classification of a protein from its primary sequence, Nucleic Acids Res. 31 (13) (2003) 3692–3697.

[21] C.Z. Cai, L.Y. Han, Z.L. Ji, Y.Z. Chen, Enzyme family classification by support vector machines, Proteins 55 (1) (2004) 66–76.

[22] I. Dubchak, I. Muchnik, S.R. Holbrook, S.H. Kim, Prediction of protein folding class using global description of amino acid sequence, Proc. Natl. Acad. Sci. U.S.A. 92 (19) (1995) 8700–8704.

[23] C. Chen, Q.M. Zhang, B. Yu, Z.M. Yu, P.J. Lawrence, Q. Ma, Y. Zhang, Improving protein-protein interactions prediction accuracy using XGBoost feature selection and stacked ensemble classifier, Comput. Biol. Med. 123 (2020), 103899.

[24] V. Saravanan, N. Gautham, Harnessing computational biology for exact linear b-cell epitope prediction: a novel amino acid composition-based feature descriptor, OMICS A J. Integr. Biol. 19 (10) (2015) 648–658.

[25] S. Kawashima, M. Kanehisa, AAindex: amino acid index database, Nucleic Acids Res. 28 (1) (2000) 374.

[26] Q.L. Xiang, F. Kaiyan, L. Bo, Y.W. Liu, G.H. Huang, Prediction of lysine malonylation sites based on pseudo amino acid, Comb. Chem. High Throughput Screen. 20 (2017) 622–628.

[27] K.C. Chou, Prediction of protein cellular attributes using pseudo-amino acid composition, Proteins 43 (3) (2001) 246–255.

[28] X. Cui, Z. Yu, B. Yu, M. Wang, B. Tian, Q. Ma, UbiSitePred: a novel method for improving the accuracy of ubiquitination sites prediction by using LASSO to select the optimal Chou`s pseudo components, Chemometr. Intell. Lab. 184 (2019) 28–43.

[29] B. Yu, Y. Zhang, X. Wang, H. Gao, J. Sun, X. Gao, Identification of DNA modification sites based on elastic net and bidirectional gated recurrent unit with convolutional neural network, Biomed. Signal Process Control 75 (2022), 103566.

[30] W. Qiu, S. Li, X. Cui, Z. Yu, M. Wang, J. Du, Y. Peng, B. Yu, Predicting protein submitochondrial locations by incorporating the pseudo position specific scoring matrix into the general Chou's pseudo-amino acid composition, J. Theor. Biol. 450 (2018) 86–103.

[31] M.M. Hasan, D.J. Guo, H. Kurata, Computational identification of protein S-sulfenylation sites by incorporating the multiple sequence features information, Mol. Biosyst. 13 (12) (2017) 2545–2550.

[32] J. Hu, X.G. Zhou, Y.H. Zhu, D.J. Yu, G.J. Zhang, TargetDBP: accurate DNA-binding protein prediction via sequence-based multi-view feature learning, IEEE ACM Trans. Comput. Biol. Bioinf 17 (4) (2020) 1419–1429.

[33] I. Fister, P.N. Suganthan, S.M. Kamal, F.M. Al-Marzouki, M. Perc, D. Strnad, Artificial neural network regression as a local search heuristic for ensemble strategies in differential evolution, Nonlinear Dynam. 84 (2) (2016) 895–914.

[34] N. Chen, W.N. Chen, J. Zhang, Fast detection of human using differential evolution, Signal Process. 110 (2015) 155–163.

[35] R. Tibshirani, Regression shrinkage and selection via the LASSO, J. Roy. Stat. Soc. B 58 (1996) 267–288.

[36] Y. Ming, L. Yi, Model selection and estimation in regression with grouped variables, J. Roy. Stat. Soc. B 68 (2006) 49–67.

[37] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A.N. Gomez, L.Kaiser, I. Polosukhin, Attention is all you need, in: Advances in Neural Information Processing Systems, 2017, pp. 5998–6008.

[38] M.T. Luong, H. Pham, C.D. Manning, Effective Approaches to Attention-Based Neural Machine Translation, 2015, 04025 arxiv Preprint. arXiv. 1508.

[39] D. Bahdanau, K. Cho, Y. Bengio, Neural Machine Translation by Jointly Learning to Align and Translate, 2014, 0473 arXiv. Preprint arXiv. 1409.

[40] S. Hochreiter, J. Schmidhuber, Long short-term memory, Neural Comput. 9 (1997), 17350.

[41] A. Graves, J. Schmidhuber, Framewise phoneme classification with bidirectional LSTM and other neural network architectures, Neural, Netw 18 (2005) 602–610.

[42] X. Wang, Y.Q. Zhang, B. Yu, A. Salhi, R. Chen, L. Wang, Z.F. Liu, Prediction of protein-protein interaction sites through eXtreme gradient boosting with kernel principal component analysis, Comput. Biol. Med. 134 (2021), 104516.

[43] X. Wang, B. Yu, A. Ma, C. Chen, B.Q. Liu, Q. Ma, Protein-protein interaction sites prediction by ensemble random forests with synthetic minority oversampling technique, Bioinformatics 35 (2019) 2395–2402.

[44] B. Yu, X. Wang, Y. Zhang, H. Gao, Y. Wang, Y. Liu, X. Gao, Rpi-Mdlstack, Predicting RNA–protein interactions through deep learning with stacking strategy and LASSO, Appl. Soft Comput. 120 (2022), 108676.

[45] Q. Wei, Q. Zhang, H. Gao, T. Song, A. Salhi, B. Yu, Deepstack-Rbp, Accurate identification of RNA-binding proteins based on autoencoder feature selection and deep stacking ensemble classifier, Knowl.-Based Syst. 256 (2022) 109875.

[46] B. Yu, W.Y. Qiu, C. Chen, A.J. Ma, J. Jiang, H.Y. Zhou, Q. Ma, SubMito-XGBoost: predicting protein submitochondrial localization by fusing multiple feature information and eXtreme gradient boosting, Bioinformatics 36 (2020) 1074–1081.

[47] J.F. Box, Gosset Guinness, Fisher, and small samples, Stat. Sci. 2 (1987) 45–52.

[48] V. Vacic, L.M. Iakoucheva, P. Radivojac, Two Sample Logo: a graphical representation of the differences between two sets of sequence alignments, Bioinformatics 22 (2006) 1536–1537.

[49] D. Jiang, W. Zang, R. Sun, Z. Wang, X. Liu, Adaptive density peaks clustering based on K-nearest neighbor and Gini coefficient, IEEE Access 8 (2020) 113900–113917.

[50] K. Chen, L.A. Kurgan, J. Ruan, Prediction of flexible rigid regions from protein sequences using k-spaced amino acid pairs, BMC Struct. Biol. 7 (2007) 25.

[51] R. Gorman, T. Sawatari, The use of multidimensional perceptual models in the selection of sonar echo features, J. Acoust. Soc. Am. 77 (1985) 1178–1184.

[52] B.C. Ross, Mutual information between discrete and continuous data sets, PLoS One 9 (2014), e87357.

[53] Q. Zou, J. Zeng, L. Cao, R. Ji, A novel features ranking metric with application to scalable visual and bioinformatics data classification, Neurocomputing 173 (2016) 346–354.

[54] L. Maaten, G. Hinton, Visualizing data using t-SNE, J. Mach. Learn. Res. 9 (2008) 2579–2605.

[55] Q. Zhang, P. Liu, Y. Han, Y. Zhang, X. Wang, B. Yu StackPDB, Predicting DNA-binding proteins based on XGB-RFE feature optimization and stacking ensemble classifier, Appl. Soft Comput. 99 (2021), 106921.

[56] Y. Freund, R.E. Schapire, A decision-theoretic generalization of on-line learning and an application to Boosting, J. Comput. Syst. Sci. 55 (1997) 119–139.

[57] N. Friedman, G. Dan, M. Goldszmidt, Bayesian network classifiers, Mach. Learn. 29 (1997) 131–163.

[58] L. Breiman, Random forest, Mach. Learn. 45 (2001) 5–32.

[59] T. Chen, C. Guestrin, XGBoost: a scalable tree boosting system, in: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2016, pp. 785–794.

[60] G.E. Hinton, R.R. Salakhutdinov, Reducing the dimensionality of data with neural networks, Science 313 (5786) (2007) 504–507.

[61] M. Wang, L. Song, Y. Zhang, H. Gao, L. Yan, B. Yu, Malsite-Deep: prediction of protein malonylation sites through deep learning and multi-information fusion based on NearMiss-2 strategy, Knowl.-Based Syst. 240 (2022), 108191.

[62] Q. Liu, F. Xia, Q.J. Yin, R. Jiang, Chromatin accessibility prediction via a hybrid deep convolutional neural network, Bioinformatics 34 (5) (2018) 732–738.

[63] S. Yadav, A. Ekbal, S. Saha, A. Kumar, P. Bhattacharyya, Feature assisted stacked attentive shortest dependency path based Bi-LSTM model for protein-protein interaction, Knowl.-Based Syst. 166 (2019) 18–29.

[64] D. Wang, D. Liu, J.C. Yu, F. He, Y. Jiang, S. Cai, J. Li, D. Xu, MusiteDeep: a deep-learning based webserver for protein post-translational modification site prediction and visualization, Nucleic Acids Res. 48 (2020) 140–146.