A hybrid deep learning framework for predicting the protein-protein interaction between virus and host

Lei Deng (leideng@csu.edu.cn)
Central South University
Wenjuan Nie
Central South University
Jiaojiao Zhao
Central South University
Jingpu Zhang
Henan University of Urban Construction

Research Article

Keywords: protein-protein interaction, Convolution neural network, LSTM

Posted Date: June 8th, 2021

DOI: https://doi.org/10.21203/rs.3.rs-506156/v1

RESEARCH

A hybrid deep learning framework for predicting the protein-protein interaction between virus and host

Lei Deng[1], Wenjuan Nie[1], Jiaojiao Zhao[1] and Jingpu Zhang[2*]

*Correspondence:
20181027@hncj.edu.cn
[2]School of Computer and Data Science, Henan University of Urban Construction, Pingdingshan, China
Full list of author information is available at the end of the article

Abstract

Background: Viral infection and diseases are caused by various viruses involved in the protein-protein interaction (PPI) between virus and host, which are a threat to human health. Studying the virus-host PPI is beneficial to apprehending the mechanism of viral infection and developing new treatment drugs. Although several computational methods for predicting the virus-host PPI have been proposed, most of them are supported by the machine learning algorithms, making the hidden high-level feature difficult to be extracted.

Results: We proposed a novel hybrid deep learning framework combined with four CNN layers and LSTM to predict the virus-host PPI only using protein sequence information. CNN can extract the nonlinear position-related features of protein sequence, and LSTM can obtain the long-term relevant information. L1-regularized logistic regression is applied to eliminate the noise and redundant information. Our model achieved the best performance on the benchmark dataset and independent set compared with other existing methods.

Conclusion: Our method, through the hybrid deep neural network, is useful for predicting virus-host PPI using protein sequence alone, and achieved the best prediction performance compared with other existing methods, which is promising on the virus-host PPI prediction

Keywords: protein-protein interaction; Convolution neural network; LSTM

Background

A variety of viruses may cause various viral diseases, such as SARS-CoV-2 that has recently raged around the world, HIV-1, Ebola, influenza virus, and so on. Although extensive research has been conducted on infectious disease, it is still regarded as a threat to human health. For example, as of March 2021, a total number of more than 128 million people have been diagnosed as COVID-19 patients caused by SARS-CoV-2 all over the world, and more than 2.8 million people died. The virus infection is mainly caused by the protein-protein interaction (PPI) between viruses and hosts, this is because the basis of communication between virus and host is formed by surface protein and molecules [1]. So, the virus-host PPI can be considered as the highly specific physical contact between host protein and virus protein, which can directly affect the viral infection of the host by regulating the host protein. Moreover, new antibacterial treatments against drug resistance are only accessible by investigating the virus-host PPI [2]. Therefore, identifying the PPI between virus and host can raise the understanding of the mechanism of viral

infection for viral defense and is also the key to developing new antiviral drugs for treatment.

Many computational methods for predicting protein-protein interactions have been proposed one after another, but most of them focused on intra-species prediction rather than inter-species. The problem of predicting virus-host interaction studied in this paper belongs to inter-species prediction, and the following methods are commonly used in the virus-host interaction prediction(see review article [3]): sequence-based [4,5], domain-based [6–8], motif-based [9–12], and three-dimensional structure-based [13, 14]. Because of the low availability of structures, methods with protein sequence information alone have been widely applied. Ranjan Kumar Barman et al. [15] integrate domain-domain association, network topology, and protein sequence information as the feature to detect PPI between virus and host by utilizing three classical machine learning algorithms: SVM, Naive Bayes, and Random Forest. Under the five-fold cross-validation, the SVM-based method achieved stable performance on the values of sensitivity and AUC. They reached 0.67 and 0.73, respectively. The protein feature was extracted by Cui et al. [4] by encoding the frequency of amino acid triplets, which was fed into SVM to identify the PPI between virus and host. This representation of feature can ignore the sequence length, generate a fixed-length feature and be extended to proteins with different types. Fatma-Elzahraa Eid et al. [16] developed Denovo based on the negative sampling of sequence and machine learning framework. Denovo can learn from the PPI of different viruses, and use the shared host protein to predict a new virus, overcoming negative interaction noise. With its high accuracy and generalization, Denovo also performs well in intra-species and bacteria-host interaction prediction. Ibrahim Ahmed et al. [1] extracted three features of amino acid quadruple, the sequence similarity of virus-host interaction pair, and human interactome graph properties, and fed them into the neural network and SVM for training. After testing and applying to human-B.anthracis prediction, the neural network model was found to outperform other methods and the SVM model. A method constructed by SVM was proposed by Zhou et al. [17] based on the protein sequence information, which is suitable for new viruses and hosts, and achieved good performance. Esmaeil Nourani et al. [18] exploited target similarity and attacker similarity which are extracted for the first time, combining with protein sequence, network topology, and gene ontology as the feature to predict the pathogens-host interaction by utilizing Bayesian matrix factorization. The model can relieve the need for negative samples, and perform better than other methods.

Nowadays, deep learning [19] has been widely applied in bioinformatics [20–23] for its superiority in capturing the inherent laws and representation levels of sample data. DPPI [24] is a deep learning framework used to predict PPI constructed by a Siamese-like convolutional neural network combined with random projection and data augmentation. This novel model is flexibly trained without significant parameters tuning, and applicable to large amounts of training data, making it more efficient on calculation and suitable for different applications. An end-to-end framework based on protein sequence information alone, PIPR, was proposed by Chen et al. [25]. PIPR employs a Siamese architecture with residual recurrent convolutional neural network and an efficient property-aware lexicon embedding approach to capture the protein sequence information, which is robust and effective. And it also has

outstanding performance, and can work on other applications, such as interaction type prediction and binding affinity estimation.

In this paper, we proposed a deep learning framework to predict the PPI between viruses and hosts, including four parts: input module, convolution module, long short-term memory network (LSTM) module, and prediction module. In the input module, we grouped amino acids into seven classes by physicochemical properties, then calculated the fixed-length feature of protein sequence, and take it as the output of the input module. The convolution module is employed to capture the local semantic association of protein sequence, and the LSTM module obtains the long-short term dependencies of protein sequence. Finally, the probability of the interaction of viruses and hosts can be obtained by the prediction module. Compared with other proposed methods used to predict PPI between viruses and hosts, our model achieved the best performance with AUC reaching 0.937. Moreover, two contributions from our model are concluded as follows: 1) Many existing methods ignore the noise of the feature itself. We applied L1-regularized logistic regression for feature selection to remove redundant or irrelevant features, thereby ensuring that the effective feature has been obtained without losing important information. 2) CNN is employed to capture the nonlinear position-related feature of protein sequence, while LSTM is applied to learn the short-term dependence of the amino acid level and the long-term dependence of the motif level. The hybrid deep neural network combined with CNN and LSTM allows our model to extract the hidden high-level features between sequences and learn the connection between features at the same time.

Methods

Datasets

The dataset we used in this paper is provided by Zhou's paper [17], which was collected from four databases: APID, IntAct, Mentha, and Uniport, and contains 12157 virus-host PPIs. Within this dataset, the hosts include 29 species from human, non-human animals, plant, bacteria, and others. The viruses contain 137 types such as Human immunodeficiency virus 1, Human SARS coronavirus, Ebola virus, Dengue virus, etc. Detailed statistics of virus-host PPIs are shown in Table 1.

Table 1 Statistics of protein-protein interaction data between viruses and hosts

Hosts classification	Virus-Host PPI	Types of interaction virus
Human	11491	246
Non-human animal	488	169
Plant	17	11
Bacteria	143	16
Others	18	15
Total	12157	457

Predicting the protein-protein interaction between virus and hosts is a classification task, so we need to learn both positive and negative samples. What's more, existing studies have shown that the best experimental effect can be achieved when the ratio of positive and negative samples is 1:1 [26]. However, the lack of standard negative samples is still a problem for interspecies PPI prediction [4, 6, 27].We randomly select 12157 pairs of data from the unknown interactions between virus and

hosts. Besides, we also use 381 pairs of PPI between the human and H1N1 virus as an independent set to verify our method.

Feature

The quality of feature representation is closely related to the performance of model. In this paper, we use the protein sequence information of viruses and hosts to extract three features of the relative frequency of amino acid triplets(RFAT), the frequency difference of amino acid triplets(FDAT), and amino acid composition(AC). Figure 1 shows the detailed process of feature extraction.

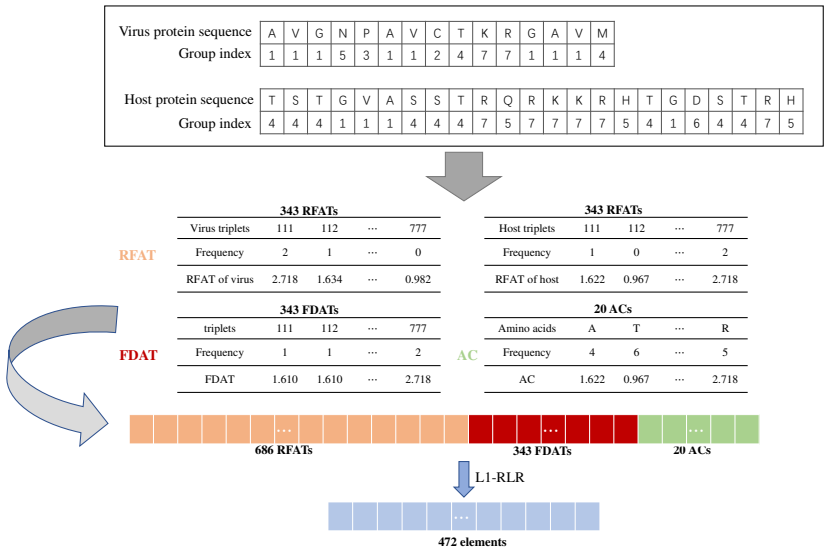

Figure 1 The extraction process of feature in the input module. We calculate three features of RFAT, FDAT, and AC from the protein sequence of virus and host, then concatenate the three features to obtain a feature vector with 1049 elements for each pair of virus-host interaction. Finally, L1-regularized logistic regression is applied for feature selection.

Feature extraction

The protein sequence is composed of 20 amino acids. Shen et al. [28] proposed a coding method based on the dipoles and volumes of the side chains to divide 20 amino acids into seven groups (Table 2), which could generate 343 ($7 \times 7 \times 7$) amino acid triplets. This coding way can significantly reduce the dimensionality of the feature vector, making the feature vector not too sparse, and also partially overcoming the overfitting problem.

Table 2 Seven groups of amino acids classified by physicochemical properties

Group1	Group2	Group3	Group4	Group5	Group6	Group7
A, G, V	C	F, I, L, P	M, S, T, Y	H, N, Q, W	D, E	K, R

The length of the protein sequence of viruses and hosts is variable, but the original variable length of protein sequence information can be mapped to a fixed-length feature vector by calculating the RFAT and FDAT. The relative frequency of amino

acid triplets (RFAT) of i-th amino acids triplets is defined by:

$$RFAT_i = e^{(f_i - avgF)/(maxF - avgF)} \tag{1}$$

where $F = f_1, f_2, \cdots, f_{343}$. In this equation, f_i indicates the frequency of $i-th$ amino acids triplets in the protein sequence of virus and host, and $avgF$ and $maxF$ represent the average frequency and the maximum frequency of all amino acids triplets in protein sequence, respectively.

For each interaction pair of virus and host, the frequency difference of amino acid triplets(FDAT) is calculated by Eq2. In a virus-host interaction pair, we define f_{vi} and f_{hi} the $i-th$ amino acid triplet frequency of the protein sequence of virus and host, respectively. $D = |f_{h1} - f_{v1}|, |f_{h2} - f_{v2}|, \cdots, |f_{h343} - f_{v343}|$, denotes the difference among the frequency of amino acid triplets in this interaction pair.

$$FDAT_i = e^{(|f_{hi} - f_{vi}| - avgD)/(maxD - avgD)} \tag{2}$$

We also calculate amino acid composition(AC), and take it as a feature of the virus-host pair. It can be defined as:

$$AC_i = \frac{f_i}{max\{f_1, f_2, \cdots, f_{20}\}} \tag{3}$$

where f_i denotes the probability of $i-th$ amino acid in the virus-host pair.

Finally, a feature vector with 1049 elements for each virus-host interaction pair could be obtained, which consists of RFAT with 686 elements (343 elements for virus and 343 elements for host), FDAT with 343 elements, and AC with 20 elements. However, the feature vector we obtained may have redundant or irrelevant features, which may cause over-fitting. Meanwhile, if the dimensionality of the feature vector is too large, the time used for calculation will increase accordingly, so feature selection is required.

Feature selection

For the advantages of easy realization, interpretability, and expansion, logistic regression has been widely used in many researches. L1-norm refers to the sum of the absolute values of each element in the weight vector w, contributing to balance the fit of training data and reduce the risk of overfitting,usually expressed as $||\omega||_i$. Add L1-norm after the loss function of the logistic regression, we could get the L1-regularized logistic regression (L1-RLR), which can avoid overfitting effectively, and is also an essential optimization problem to minimize the following loss function:

$$J(\omega) = C\{\sum_{i=0}^{n} -y_i log h_\omega(x_i) - (1 - y_i) log(1 - h_\omega(x_i))\} + \alpha||\omega||_i \tag{4}$$

where C represents the penalty term, $\alpha||\omega||_i$ is the regularization term, and α is the regularization parameter, which could restrict some parameters in the loss function.

So we utilized L1-RLR, also called Lasso regression, for feature selection. Then we obtain a feature vector with 472 elements for each virus-host interaction pair.

Model

We proposed a novel model integrating deep neural network and protein sequence information to predict the protein-protein interaction between viruses and hosts, including input module, convolution module, long short-term memory network, and prediction module. The architecture of our model is shown in Figure 2.

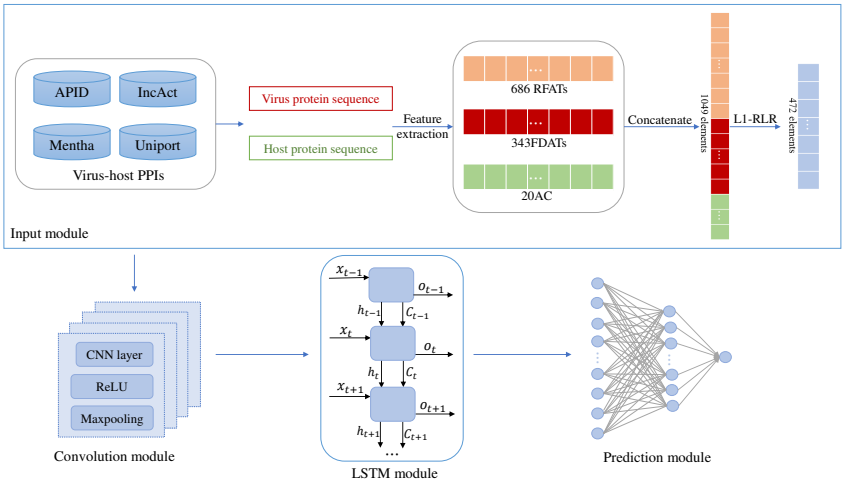

Figure 2 Architecture of the novel hybrid neural network model for virus-host PPI prediction. First, we encode the amino acid sequence of protein and calculate three features of RFAT, FDAT, and AC. Next, the combined features selected by L1-regularized logistic regression are fed into convolution module, LSTM module and two fully connected layers. Finally, the output from the connected layer is fed into a sigmoid activation function to predict the probability of the protein-protein interaction between virus and host.

Input module

The input module is mainly used to generate feature vectors that can be input into the deep neural network for training. We extracted RFAT, FDAT, and AC for each virus-host pair based on the protein sequence of virus and host. Concatenating the three features, we could obtain a feature vector with 1049 elements($343 \times 2 + 343 + 20$). Then, L1-RLR is applied for feature selection. So the output of the input module is a 472-dimensional feature vector, which is also the input of our deep learning framework.

Convolution module

Convolution neural network is a feed-forward network with convolution calculations and deep structure [29]. It was discovered by Hubel and Wiesel [30] in the 1960s, and has been applied widely in bioinformatics currently. With the two characteristics of local perception and parameter sharing, the convolution neural network can reduce the complexity of the framework, and decrease the number of weights while also ensuring that the convolution kernels have the strongest response to the local mode of the input.

This module consists of four layers of convolution neural network. Each layer of CNN includes three parts: convolution layer, rectified linear unit (ReLU), and

pooling. The convolution layer mainly performs convolution operations while simulating cells with local receptive fields in the brain using local connections and weight sharing to extract hidden features in the sequence. ReLU is an activation function that changes the negative values output from the convolutional layer to zero while the positive values remain unchanged. The pooling layer mainly performs down-sampling operations such as maximum pooling and average pooling. After the input data is down-sampled by the pooling layer, the output data matrix will be smaller, but the number remains the same. Therefore, the pooling layer can compress the data output from the previous layer to reduce the computational complexity, decrease the number of learning parameters and avoid over-fitting effectively. Then, the dimension of feature has been reduced by the maximum pooling of the output of the ReLU layer. So, the output of each layer of CNN is calculated by:

$$O_c = MaxPooling_s(ReLU(Conv_{k,l}(I))) \tag{5}$$

where I is the input vector of this module and O_c is the output vector. $Conv_{k,l}()$ is the convolution operation on the input file by using k filters with the filter length of l. The value of k of the four CNNs is set as 128, 64, 32, and 16, while the value of l is set as 15, 10, 8, and 5, respectively. $Maxpooling_s(\cdot)$ performs the maximum pooling on the output of the ReLU layer, where s is set as 2.

Long short-term memory network

Long short-term memory network(LSTM) is a special recurrent neural network that can avoid gradient disappearance and gradient explosion in the training process of long sequences [31]. Moreover, LSTM can effectively solve the long dependence problem in long sequences, such as protein sequence.

The key to LSTM is the state of the cell. The cell state is like a conveyor belt running directly on the entire chain, with only a small amount of linear interaction, thereby ensuring that the information flows through the entire network structure and remains unchanged. LSTM removes or adds information to the cell state through a well-designed structure called 'gate'. Gate controls the circulation and the loss of the feature, allowing information to pass through selectively. There are three gates in LSTM to protect and control the state of the cell, namely, forget gate, input gate, and output gate. The working flow of LSTM is described as follows:

- The previous hidden state of LSTM h_{t-1} and the current input x_t go through the forget gate to determine what information of the previous cell state C_{t-1} can be retained to the current state C_t:

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f) \tag{6}$$

- The last hidden state h_{t-1} and the current input x_t go through a neural network($tanh$ layer) and the input gate to obtain the update state of the current cell $\tilde{C}_t$ and the input information, respectively:

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i) \tag{7}$$

$$\tilde{C}_t = tanh(W_C \cdot [h_{t-1}, x_t] + b_C) \tag{8}$$

- The forget gate controls what information of the previous cell state C_{t-1} will be added to the current cell state C_t, while the input gate controls what information of the update state of the current cell $\tilde{C}_t$ will be added to the current cell state C_t:

$$C_t = f_t * C_{t-1} + i_t * \tilde{C}_t \tag{9}$$

- Determine what information of the current cell state C_t will be output to the hidden state h_t, which can be obtained by the current cell state C_t and the output gate.

$$o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o) \tag{10}$$

$$h_t = o_t * tanh(C_t) \tag{11}$$

where σ and $tanh$ are the activation function, representing the Sigmoid function and tanh function, respectively. W and b are the weights and bias of the activation function.

Table 3 The parameters and output size of each module

Module	Layer	Parameters	Output_size
Input Module	Input	Batch_size=16	(16,1472)
Convolution Module	Convolution Layer 1	Filters =1 Filter_length=15 Padding=8 Activation=ReLU	(16,1474)
	Maxpooling	Pooling_size=2	(16,1237)
	Convolution Layer 2	Filters =1 Filter_length=10 Padding=8 Activation=ReLU	(16,1244)
	Maxpooling	Pooling_size=2	(16,1122)
	Convolution Layer 3	Filters =1 Filter_length=8 Padding=8 Activation=ReLU	(16,1131)
	Maxpooling	Pooling_size=2	(16,166)
	Convolution Layer 4	Filters =1 Filter_length=5 Padding=8 Activation=ReLU	(16,178)
	Maxpooling	Pooling_size=2	(16,139)
LSTM Module	LSTM layer	Hidden_size=80 Bidirectional=True	(16,80)
Prediction Module	Fully connected layer	Number of neurons=64 Dropout ratio=2 Activation=sigmoid	(64,1)

Prediction module

The prediction module is composed of two fully connected layers. The hidden feature can be extracted from original sample data through the convolution module and LSTM. And the fully connected layer maps the distributed feature representation

learned in the previous operation to the sample label space. Furthermore, to reduce the risk of over-fitting and improve the generalization ability of model, the dropout layer is used as a regularization technique between the two fully connected layers [32]. Finally, sigmoid is selected as the activation function to map the output into $[0, 1]$. The output of the prediction module can be calculated by:

$$O_p = Sigmoid(Linear_b(Dropout_r(Linear_a(I_p)))) \tag{12}$$

where O_p denotes the output of the prediction module, indicating the probability of the interaction between virus and host. The input of the prediction module is expressed as I_p, which is also the output of LSTM. The number of neurons in the two fully connected layers is set as a and b, respectively, while r is a hyper-parameter that is selected through experiments. The paramters of each layer of our network are shown in Table 3.

Finally, we use the binary cross-entropy as the loss function as shown in Eq13, which is applied to the measurement of the degree of matching between the model and the experimental verification data. Adam is employed to update the weights of networks iteratively.

$$loss(p, c) = -(clog(p) + (1 - c)log(1 - p)) \tag{13}$$

where c is the true label and p is the predicted value of our model.

Results and discussion

Performance evaluation metrics

To evaluate the performance of our method, we use several well-recognized performance metrics, such as accuracy(ACC), precision, recall, specificity, F1-socre, MCC, and AUC. The definition of these performance metrics is summarized in Table 4.

Table 4 Performance evaluation metrics

Performance metrics	Definition
ACC	$\frac{TP+TN}{TP+TN+FP+FN}$
Precision	$\frac{TP}{TP+FP}$
Recall	$\frac{TP}{TP+FN}$
Specificity	$\frac{TN}{TN+FP}$
F1_Score	$\frac{2\times Precision\times Recall}{Precision+Recall}$
MCC	$\frac{TP\times TN-FP\times FN}{\sqrt{(TP+FP)(TP+FN)(TN+FP)(TN+FN)}}$
AUC	The area under the ROC curve

where TP, TN, FP, and FN are the numbers of true-positive(the interaction host proteins are correctly identified as the interaction with viruses), true-negative (the non-interaction host proteins are correctly identified as the non-interaction with viruses), false-positive (the non-interaction host proteins are identified as the interaction with viruses), false-negative (the interaction host proteins are identified as the non-interaction with viruses), respectively.

Comparison with different combination of features

We have extracted three features of RFAT, FDAT, and AC from the protein sequence of virus and host. To study the impact of different features on the performance of our model, five different feature combinations are fed into five-fold cross-validation, and their prediction performance is shown in Table 5. It is apparent that RFAT performs better than FDAT and AC, with its AUC and MCC higher than the other two by 16% and 11.3%, and 34.9% and 29.5%, respectively. When we combine the three features instead of using a single feature, the prediction performance of the model is significantly improved. What's more, the prediction performance of the model is the best when L1-RLR is applied for the selection of the combined feature, with ACC, AUC, and MCC reaching 89.28%, 95.2%, and 76.9%, that is, 1.44%, 1.2%, and 1.2% higher than the case without L1-RLR. This result also indicates that L1-RLR plays an important role in feature extraction, and effectively eliminates the noise in the feature itself.

Table 5 Results of predicting virus-host PPI using different combinations of features

Features	ACC	Precision	Recall	Specificity	MCC	AUC
RFAT	0.8761	0.8792	0.8255	0.8790	0.753	0.925
FDAT	0.6807	0.6847	0.6994	0.6962	0.404	0.765
AC	0.7237	0.7952	0.6994	0.7671	0.458	0.812
RDFA+FDAT+AC	0.8784	0.8774	0.8798	0.8918	0.757	0.940
RFAT+FDAT+AC+L1-RLR	0.8928	0.9013	0.9032	0.8965	0.769	0.952

Comparison with different model network

Our framework of the network consists of four CNN and LSTM. The convolution module is used to capture the latent nonlinear position-related feature in protein sequence to enhance the high correlation of protein interaction, and LSTM is applied to extract long-term relevant information of the protein sequence. To evaluate the importance of each module in our model, we compared our model (4CNN+LSTM) with three different frameworks: CNN only, LSTM only, and 3CNN+LSTM (consisting of LSTM and convolution module with three CNN layers). The prediction effects of the above four different frameworks are shown in Figure 3. It is apparent that the framework of our model shows the best performance, of which the AUC value is 1.02, 1.09, and 1.12 times of that of the other three frameworks, indicating the effectiveness and superiority of our model.

Comparison with different parameter selection

In a deep neural network, the more different hyperparameters we choose, the more different prediction performance of the model we will obtain. So we need to find the optimal hyperparameters of the model to train the model more scientifically, and improve resource utilization. Batch_size is the number of samples sent to the neural network during each training epoch, which affects the speed and optimization of the model. Increasing the batch_size could improve the parallelization efficiency, raise memory utilization, and reduce training shock. If the value of batch_size is set to be too large, it may cause insufficient memory or program kernel crash because of the restriction of memory resources. So we search the optimal value of batch_size at 8, 16, 32, 64, and 128. As shown in Figure 4(a), when the batch_size is set as

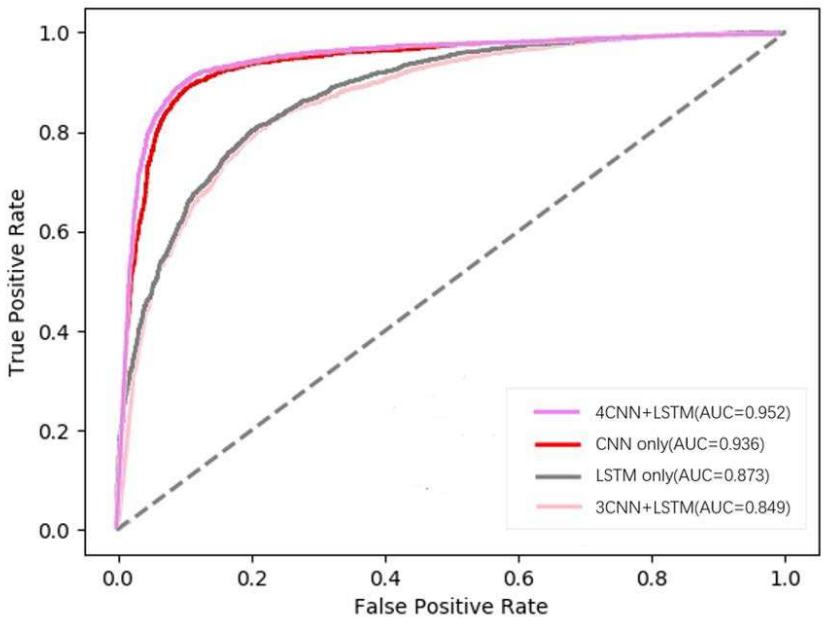

Figure 3 Performance comparison with different model framework for virus-host PPI prediction. For fairness, the CNN only and LSTM only are constructed by the same parameters and architecture of the original structure. The 3CNN+LSTM consists of 64, 32, and 16 filters in each layer with the filter size of 10, 8, and 5 respectively.

16, the performance of the model is the best. Learning rate is the magnitude of the updated network weight in the optimization algorithm. Excessive learning rate may cause the model to fail to converge, and the loss will oscillate up and down. On the contrary, if the learning rate is too low, the model will converge slowly and make the calculation time longer. We search the optimal value of learning rate at 0.001, 0.0005, 0.0001, 0.00005, and 0.00001. The performance is the best when the learning rate is set as 0.0001, as shown in Figure 4(b).

Comparison with other existing method

We compared our method with other proposed methods used to predict the virus-host interaction, including Zhou's method [17], Ahmed's method [1], and Denovo [16], based on the independent dataset and the five-fold cross-validation of benchmark dataset. With the result of comparison shown in Table 6, the prediction performance of our method is obviously higher than that of the other three methods both on the independent dataset and five-fold cross-validation. The AUC value of our method reached 0.952, while that of Denovo, Ahmed's method, and Zhou's method only reached 0.876, 0.911, and 0.92 on the five-fold cross-validation, respectively. In addition, our method also achieved the best prediction performance on the independent test set, with the AUC value of 0.937.

Conclusion

In the research of identifying protein-protein interaction, most methods proposed tend to predict the PPI within the same species instead of crossing different species.

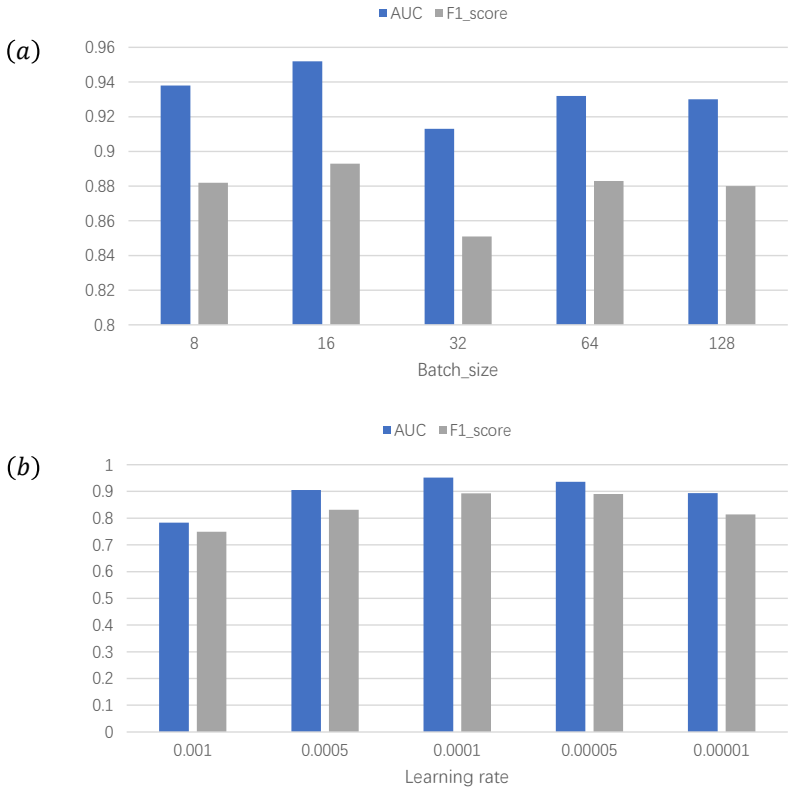

Figure 4 (a) The influence of the virus-host PPI prediction performance with different batch_size. (b) The influence of the virus-host PPI prediction performance at different learning rate.

Table 6 The AUC performance of virus-host PPI prediction compared with other existing methods

Methods / Dataset	Our method	Denovo	Ahmed's method	Zhou's method
5-fold cross-validation on benchmark dataset	**0.952**	0.876	0.911	0.92
Independent dataset	**0.937**	0.836	0.876	0.886

Because of the difficulty of intra-species prediction to distinguish the protein-protein interaction between the same species and different species, the methods applied for intra-species prediction are not applicable for inter-species prediction. However, it is necessary to learn the interaction between different species. For example, predicting the interaction between viruses and hosts is beneficial for understanding viral infections and designing new antibacterial drugs for treatment.

In this paper, we devoted ourselves to the understanding of the protein-protein interaction between virus and host, and a deep learning framework has been developed to predict virus-host PPI. Three features of RFAT, FDAT, and AC are extracted from protein sequence to constitute the feature vector which transforms the original variable sequence information to the fixed-length feature information. We also verified that the combination of the three features was more effective than using a single feature. The performance was significantly improved when the L1-regularized

logistic regression was used to eliminate the redundant or irrelevant features. CNN is good at extracting latent nonlinear position-related features in protein sequences. Simultaneously, LSTM performs well in learning the amino acid's short-term relevant information and motif's long-term relevant information. So we designed a hybrid neural network that combined a convolution module with four CNN and LSTM to predict virus-host PPI, which performs better than other machine learning algorithms or a single neural network only. All in all, our method based on the hybrid deep neural network is a promising calculation method that can predict virus-host PPI effectively by using the protein sequence only and achieves the best prediction performance than other existing methods.

Declarations

Acknowledgements
Not applicable.

Funding
This work was supported by the No.61972422 grants from National Natural Science Foundation of China. Publication costs are funded by the No.61972422 grant from National Natural Science Foundation of China. The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

Abbreviations
PPI: protein-protein interaction
CNN: convolutional neural network
LSTM: Long short-term memory network
L1-RLR: L1-regularized logistic regression
RFAT: the relative frequency of amino acid triplets
FDAT: the frequency difference of amino acid triplets
AC: amino acids composition

Availability of data and materials
Not applicable.

Ethics approval and consent to participate
Not applicable.

Competing interests
The authors declare that they have no competing interests.

Consent for publication
Not applicable.

Authors' contributions
LD and JJZ designed and implemented the prediction model. JJZ and WJN wrote the manuscript. JPZ assisted the work and examined the results. WJN revised the manuscript. All authors read and approved the final manuscript.

Author details
[1]School of Computer Science and Engineering, Central South University, Changsha, China. [2]School of Computer and Data Science, Henan University of Urban Construction, Pingdingshan, China.

References

1. Ahmed, I., Witbooi, P., Christoffels, A.: Prediction of human-bacillus anthracis protein–protein interactions using multi-layer neural network. Bioinformatics **34**(24), 4159–4164 (2018)
2. Nourani, E., Khunjush, F., Durmuş, S.: Computational approaches for prediction of pathogen-host protein-protein interactions. Frontiers in microbiology **6**, 94 (2015)
3. Zheng, N., Wang, K., Zhan, W., Deng, L.: Targeting virus-host protein interactions: Feature extraction and machine learning approaches. Current drug metabolism **20**(3), 177–184 (2019)
4. Cui, G., Fang, C., Han, K.: Prediction of protein-protein interactions between viruses and human by an svm model. In: BMC Bioinformatics, vol. 13, pp. 1–10 (2012). Springer
5. Kim, B., Alguwaizani, S., Zhou, X., Huang, D.-S., Park, B., Han, K.: An improved method for predicting interactions between virus and human proteins. Journal of bioinformatics and computational biology **15**(01), 1650024 (2017)

6. Dyer, M.D., Murali, T., Sobral, B.W.: Computational prediction of host-pathogen protein–protein interactions. Bioinformatics **23**(13), 159–166 (2007)
7. Dyer, M.D., Murali, T., Sobral, B.W.: Supervised learning and prediction of physical interactions between human and hiv proteins. Infection, Genetics and Evolution **11**(5), 917–923 (2011)
8. Zheng, L.-L., Li, C., Ping, J., Zhou, Y., Li, Y., Hao, P.: The domain landscape of virus-host interactomes. BioMed research international **2014** (2014)
9. Evans, P., Dampier, W., Ungar, L., Tozeren, A.: Prediction of hiv-1 virus-host protein interactions using virus and host sequence motifs. BMC medical genomics **2**(1), 1–13 (2009)
10. Segura-Cabrera, A., García-Pérez, C.A., Guo, X., Rodríguez-Pérez, M.A.: A viral-human interactome based on structural motif-domain interactions captures the human infectome. PloS one **8**(8), 71526 (2013)
11. Becerra, A., Bucheli, V.A., Moreno, P.A.: Prediction of virus-host protein-protein interactions mediated by short linear motifs. BMC bioinformatics **18**(1), 1–11 (2017)
12. Zhang, A., He, L., Wang, Y.: Prediction of gcrv virus-host protein interactome based on structural motif-domain interactions. BMC bioinformatics **18**(1), 1–13 (2017)
13. Doolittle, J.M., Gomez, S.M.: Structural similarity-based predictions of protein interactions between hiv-1 and homo sapiens. Virology journal **7**(1), 1–15 (2010)
14. De Chassey, B., Meyniel-Schicklin, L., Aublin-Gex, A., Navratil, V., Chantier, T., André, P., Lotteau, V.: Structure homology and interaction redundancy for discovering virus–host protein interactions. EMBO reports **14**(10), 938–944 (2013)
15. Barman, R.K., Saha, S., Das, S.: Prediction of interactions between viral and host proteins using supervised machine learning methods. PloS one **9**(11), 112034 (2014)
16. Eid, F.-E., ElHefnawi, M., Heath, L.S.: Denovo: virus-host sequence-based protein–protein interaction prediction. Bioinformatics **32**(8), 1144–1150 (2016)
17. Zhou, X., Park, B., Choi, D., Han, K.: A generalized approach to predicting protein-protein interactions between virus and host. BMC genomics **19**(6), 69–77 (2018)
18. Nourani, E., Khunjush, F., Sevilgen, F.E.: Virus–human protein–protein interaction prediction using bayesian matrix factorization and projection techniques. Biocybernetics and Biomedical Engineering **38**(3), 574–585 (2018)
19. LeCun, Y., Bengio, Y., Hinton, G.: Deep learning. nature **521**(7553), 436–444 (2015)
20. Li, H., Gong, X.-J., Yu, H., Zhou, C.: Deep neural network based predictions of protein interactions using primary sequences. Molecules **23**(8), 1923 (2018)
21. Kong, Y., Yu, T.: forgenet: a graph deep neural network model using tree-based ensemble classifiers for feature graph construction. Bioinformatics **36**(11), 3507–3515 (2020)
22. Shi, Q., Chen, W., Huang, S., Jin, F., Dong, Y., Wang, Y., Xue, Z.: Dnn-dom: predicting protein domain boundary from sequence alone by deep neural network. Bioinformatics **35**(24), 5128–5136 (2019)
23. Deng, L., Liu, Y., Shi, Y., Liu, H.: A deep neural network approach using distributed representations of rna sequence and structure for identifying binding site of rna-binding proteins. In: 2019 IEEE International Conference on Bioinformatics and Biomedicine (BIBM), pp. 12–17 (2019). IEEE
24. Hashemifar, S., Neyshabur, B., Khan, A.A., Xu, J.: Predicting protein–protein interactions through sequence-based deep learning. Bioinformatics **34**(17), 802–810 (2018)
25. Chen, M., Ju, C.J.-T., Zhou, G., Chen, X., Zhang, T., Chang, K.-W., Zaniolo, C., Wang, W.: Multifaceted protein–protein interaction prediction based on siamese residual rcnn. Bioinformatics **35**(14), 305–314 (2019)
26. Mei, S., Zhu, H.: Computational reconstruction of proteome-wide protein interaction networks between htlv retroviruses and homo sapiens. BMC bioinformatics **15**(1), 1–10 (2014)
27. Tastan, O., Qi, Y., Carbonell, J.G., Klein-Seetharaman, J.: Prediction of interactions between hiv-1 and human proteins by information integration. In: Biocomputing 2009, pp. 516–527. World Scientific, ??? (2009)
28. Shen, J., Zhang, J., Luo, X., Zhu, W., Yu, K., Chen, K., Li, Y., Jiang, H.: Predicting protein–protein interactions based only on sequences information. Proceedings of the National Academy of Sciences **104**(11), 4337–4341 (2007)
29. Goodfellow, I., Bengio, Y., Courville, A., Bengio, Y.: Deep Learning vol. 1. MIT press Cambridge, ??? (2016)
30. Hubel, D.H., Wiesel, T.N.: Receptive fields, binocular interaction and functional architecture in the cat's visual cortex. The Journal of physiology **160**(1), 106–154 (1962)
31. Huang, Z., Xu, W., Yu, K.: Bidirectional lstm-crf models for sequence tagging. arXiv preprint arXiv:1508.01991 (2015)
32. Zou, Q., Xing, P., Wei, L., Liu, B.: Gene2vec: gene subsequence embedding for prediction of mammalian n6-methyladenosine sites from mrna. Rna **25**(2), 205–218 (2019)