LPI-CSFFR: Combining Serial Fusion with Feature Reuse for Predicting LncRNA-Protein Interactions

This paper was downloaded from TechRxiv (<https://www.techrxiv.org>).

LICENSE

CC BY 4.0

SUBMISSION DATE / POSTED DATE

24-01-2022 / 28-01-2022

CITATION

Tan, Jianjun; Huang, Xiaoqian; Shi, Yi; Yan, Jing; Qu, Wenyan; Li, Xiaoyi (2022): LPI-CSFFR: Combining Serial Fusion with Feature Reuse for Predicting LncRNA-Protein Interactions. TechRxiv. Preprint.
<https://doi.org/10.36227/techrxiv.18973652.v1>

DOI

[10.36227/techrxiv.18973652.v1](https://doi.org/10.36227/techrxiv.18973652.v1)

LPI-CSFFR: Combining Serial Fusion with Feature Reuse for Predicting LncRNA-Protein Interactions

Xiaoqian Huang, Yi Shi, Jing Yan, Wenyan Qu, Xiaoyi Li, and Jianjun Tan

Abstract—Long non-coding RNAs (LncRNAs) play important roles in a series of life activities, and they function primarily with proteins. The wet experimental-based methods in lncRNA-protein interactions (lncRPIs) study are time-consuming. Therefore, in this study, we propose a reliable computational-based method called LPI-CSFFR using sequences, secondary structures, and physicochemical properties of proteins and lncRNAs for training and predicting lncRPIs. Combining Serial Fusion with Feature Reuse is utilized based on the deep learning convolution neural network (CNN) algorithm. The experimental results indicate that LPI-CSFFR achieves superior performance on benchmark datasets RPI1460 and RPI1807 with an accuracy of 83.1% and 98%, respectively. We further compare LPI-CSFFR with the state-of-the-art existing methods on the same benchmark datasets to evaluate the performance. In addition, we predict the independent RPI9373 dataset using the model trained on RPI1460. The results show that LPI-CSFFR is promising for predicting lncRPIs. The source code of LPI-CSFFR and the datasets used in this study are available at <https://github.com/JianjunTan-Beijing/LPI-CSFFR>.

Index Terms—Convolution neural network, Serial fusion, Feature reuse, LncRNA-Protein interactions;

Manuscript received January 23, 2022. This work was supported by the Beijing Natural Science Foundation under Grant 2202002 and the Chinese Natural Science Foundation project under Grant 21173014. (Corresponding author: Jianjun Tan.)

Xiaoqian Huang is with Department of Biomedical Engineering, Faculty of Environment and Life, Beijing University of Technology, Beijing, China (e-mail: huangxq@emails.bjut.edu.cn).

Yi Shi is with Department of Biomedical Engineering, Faculty of Environment and Life, Beijing University of Technology, Beijing, China (e-mail: shiyi_email@emails.bjut.edu.cn).

Jing Yan is with Department of Biomedical Engineering, Faculty of Environment and Life, Beijing University of Technology, Beijing, China (e-mail: yanjing@emails.bjut.edu.cn).

Wenyan Qu is with Department of Biomedical Engineering, Faculty of Environment and Life, Beijing University of Technology, Beijing, China (e-mail: quwenyan@emails.bjut.edu.cn).

Xiaoyi Li is with Department of Biomedical Engineering, Faculty of Environment and Life, Beijing University of Technology, Beijing, China (e-mail: lixiaoyi330@emails.bjut.edu.cn).

Jianjun Tan is with Department of Biomedical Engineering, Faculty of Environment and Life, Beijing University of Technology, Beijing International Science and Technology Cooperation Base for Intelligent Physiological Measurement and Clinical Transformation, Beijing, China (e-mail: tanjianjun@bjut.edu.cn).

I. INTRODUCTION

NON-CODING RNAs (ncRNAs) play significant functions in fundamental cellular processes on a genome-wide scale such as epigenetic regulations [1], chromatin remodeling [2], the process of transcription [3], and post-transcriptional processing [4]. Long ncRNAs (lncRNAs) with a length of more than 200 nucleotides are a heterogeneous class of ncRNAs. They function by interacting with RNA binding protein (RBP) molecules to work in biological life activities [5]–[7], especially in the initiation and progression of neurological diseases, adipocyte metabolism-related diseases, breast cancer, and so on [8]–[10]. Therefore, identifying potential lncRNA-protein interactions (lncRPIs) is of great importance for exploring their intrinsic molecular mechanisms. At present, the identification of ncRPIs methods is usually divided into two classes: the experimental methods and the computational methods. The experimental methods are often time-consuming and labor-intensive [11], while the computational methods for predicting ncRPIs are not subject to these constraints.

Recently, with the development of next-generation sequencing technologies and the rapid increase of wet experimental data [12], a large number of promising advances have been made from the lncRNA and amino acid feature information to predict the ncRPIs [13]–[16]. Muppirala *et al.* [17] proposed RPISeq using sequence information for the prediction of RNA-protein interactions (RPIs) based on the Support Vector Machine (SVM) classifier and the Random Forest (RF) classifier. Lu *et al.* [18] created a framework named lncPro, analyzed RNA and protein secondary structures together with hydrogen bonding and van der Waals interactions among Amino Acids for the ncRPI prediction based on Fisher's linear discriminant approach. IPMiner [19], a model learned high-level representations from RNA and protein sequences with stacked autoencoder (SAE) to achieve accurate prediction. Dai *et al.* [20] proposed CFRP by constructing complex features for prediction based on the RF classifier. Peng *et al.* [16] put forward RPITER with a hierarchical deep learning framework, which fed the sequence and secondary structure of RNA and protein by improved conjoint triad feature (CTF) to the Convolutional Neural Network (CNN) and SAE. Wang *et al.* [21] extracted several feature information descriptors and designed EDLMFC, combining CNN with Bi-Directional Long Short-Term Memory network (BLSTM) to predict ncRPIs. Compared to traditional machine learning-based methods such as SVM and RF, deep learning-based methods are superior in

classification predictions, especially when the amount of data is huge and data characteristics are not carefully designed [13].

In the present work, we propose a feature reuse-based approach named LPI-CSFFR using CNN connecting all characteristics in serial fusion. Firstly, we collect data from the PDB [22] and NPInter v2.0 [23] database, which are grouped into training datasets (RPI1460, RPI1807) and independent test dataset (RPI9373) according to the database sources. Secondly, we convert lncRNA and protein sequences, secondary structures, and physicochemical properties into numerical vectors that are easily distinguished by the computer. Then, we combine the protein and lncRNA vectors into the embedding matrix $Matrix^i$, where i means one of the three features described above. Finally, we apply a tandem framework with feature reuse referred as to CSFFR_SER based on CNN to extract the high-level representations from these numerical vectors. In order to test the framework performance of the serial fusion, we also design a parallel fusion framework for comparison. When constructing the model, we adopt 5-fold cross validation and evaluate the performance of the LPI-CSFFR in different data sets. In addition, we have also compared with other state-of-the-art methods. Good integrated results show that the LPI-CSFFR is a promising method for predicting the interaction of lncRNA-protein. The flowchart of the proposed method is shown in Fig. 1. In the $Matrix^i$, p and r are used to refer to the protein and lncRNA respectively, and V is the value of numeric vectors, k the number of samples, m the dimension of protein feature vectors, n the dimension of lncRNA feature vectors.

II. MATERIAL AND METHODOLOGY

A. Benchmark datasets

Deep learning usually requires both positive and negative samples for training. To validate the performance of the proposed method LPI-CSFFR, we construct a dataset of lncRNA-protein interactions containing 1460 proteins and 291 lncRNAs, named RPI1460, with 1460 positive pairs (i.e. lncRNA-protein interactive pairs) and 1460 negative pairs (i.e. lncRNA-protein non-interactive pairs). The positive pairs came from three ncRNA-protein interaction datasets of RPI488 [19], RPI2241 [17], and RPI1807 [14] which were built by calculating the atom distance between RNA and protein in RNA-protein complexes from the Protein Data Bank (PDB) database [22]. We remove ncRNA sequences less than or equal to 200nt and interactive pairs duplicated in the above three datasets, and obtain 1460 positive samples. Meanwhile, we collect the RPI1807 dataset in the same way including 1152 positive and 1215 negative pairs from the original literature that it was constructed based on the atom distance threshold values completely including negative samples. To test the robustness of LPI-CSFFR, we also collect an independent test set from the previous study of Pan *et al* [19]. After removing ncRNA sequences not more than 200nt, we name it as RPI9373, which only contained 9373 positive pairs including 6 organisms.

Fig.1 Flowchart of the proposed method LPI-CSFFR.

1) Reliable Negative Samples Construction

Due to the scarcity of threshold-based negative samples, researchers have often randomly selected shuffled positive pairs as negative samples, which might lead to large numbers of false-negative samples. To construct reliable negative samples for RPI1460, we use the following criteria from Cheng *et al.* [24], instead of randomly extracting lncRNA and protein negative pairs which might get many false-negative pairs.

Firstly, we calculate a lncRNA-protein pair interaction score I based on the known lncRNA-protein interactive pairs in the positive samples, and sort the I for lncRNA-protein pairs by ascending order. For any protein-lncRNA pair P_i - R_j , the interaction score is defined by $I(P_i, R_j)$:

Then, considering the comprehensiveness of the negative sample coverage, we took the random selection of negative

$$I(P_i, R_j) = \begin{cases} 1 & (P_i, R_j) \in pos \\ \sum_{k=1}^n D(P_k, R_j) S(P_i, P_k) & (P_i, R_j) \notin pos, k \neq i \end{cases} \quad (1)$$

$$D(P_k, R_j) = \begin{cases} 1 & (P_k, R_j) \in pos \\ 0 & (P_k, R_j) \notin pos \end{cases} \quad (2)$$

$$S(P_i, P_k) = \frac{sw(P_i, P_k)}{\sqrt{sw(P_i, P_i)sw(P_k, P_k)}} \quad (3)$$

Where pos is positive pairs, n is the protein number of positive samples, $S(P_i, P_k)$ is the sequence similarity score of proteins P_i and P_k based on the Smith-Waterman algorithm [25], and $sw(., .)$ denotes the original Smith-Waterman score of protein P_i and P_k .

sample pairs at a certain I threshold (i.e. 0.4). When interaction score I of lncRNA-protein pairs is less than 0.4, the negative samples include all 1460 proteins and 291 lncRNAs.

B. The feature encoding of proteins and RNAs

In the literature of lncRNA-protein interaction recognition, there is a tendency to integrate sequence or/and structural profiles as methodic features into the deep learning models to well predict unknown lncRNA-protein interactions, such as IPMiner [19], RPITER [16], and EDLMFC [21]. We also extract physicochemical information from the study of Liu *et al.* [26] in addition to taking the original sequence and secondary structural features.

1) Original sequences encoding

To input RNA and protein sequences into deep learning models, the data must first be transformed into numerical representations. Because the RNA and protein sequence lengths vary in a large range in our datasets, the commonly used fixed length encoding methods, such as one-hot encoding [13], do not fit our problem. Therefore, we adopt the sequence frequency encoding method from Peng *et al.* [16], which counted l - K -mer frequency for more information in a sequence to form fixed length vector representation.

Different from the previous coding way is that we use original sequences, instead of classifying 20 amino acids into seven groups based on their physicochemical properties of dipole moments and side chain volumes [14], [16], [19]. We consider the physicochemical properties of amino acids into one of the protein features, so it is no longer classified in the

sequence encoding stage. Each protein sequence is represented by a twenty-letter alphabet. Considering a protein sequence $P = 'P_1P_2...P_N'$, where $P_i \in \{A, C, D, E, F, G, H, I, K, L, M, N, P, Q, R, S, T, V, W, Y\}$, $i = 1, 2, \dots, N$, and m represents the length of the protein sequence. Here $K = 3$, we consider the one, two, and three letters into a 1-tuple, 2-tuple, and 3-tuple separately, and use them to scan the one, two, and three adjacent letters in turn and form the protein sequences as follows: (1), (2), ..., (N), (1, 2), (2, 3), ..., (N-1, N), (1, 2, 3), (2, 3, 4), ..., (N-2, N-1, N). Then we get an $8420 \times (3 \times N - 3)$ ($\sum_{i=1}^N 120^i = 8420$) adjacency matrix, which is a one-hot encoding matrix. To fix the length of the eigenvector, we make a sum of the frequencies and divide it by the corresponding number of scans for each tuple as standardization. Now we obtain an 8420×1 matrix for a protein sequence. According to this calculation, we obtain an $8420 \times k$ matrix Ps transformed with k protein sequences, similarly, a $340 \times k$ ($\sum_{i=1}^N 14^i = 340$) matrix Rs transformed with k RNA sequences when $K = 4$ from k sample pairs. We splice the RNA and protein sequence vectors and form one $(8420 + 340) \times k$ matrix Seq_k .

2) Secondary structures encoding

The lncRNA secondary structures are predicted by RNAfold

$$Seq_{(8420+340) \times k} = Ps_{8420 \times k} \oplus Rs_{340 \times k} \quad (4)$$

$$Ps_{8420 \times k} = Ps_{8420 \times 1}^1 \oplus Ps_{8420 \times 1}^2 \oplus \dots \oplus Ps_{8420 \times 1}^k \quad (5)$$

$$Rs_{340 \times k} = Rs_{340 \times 1}^1 \oplus Rs_{340 \times 1}^2 \oplus \dots \oplus Rs_{340 \times 1}^k \quad (6)$$

$$Ps_{8420 \times 1} = \sum_{h=1}^{3 \times M - 3} (p_{1,h})_{1 \times (3 \times M - 3)} \quad (7)$$

$$\oplus \sum_{j=1}^{3 \times M - 3} (p_{2,h})_{1 \times (3 \times M - 3)} \oplus \dots \oplus \sum_{j=1}^{3 \times M - 3} (p_{8420,h})_{1 \times (3 \times M - 3)}$$

$$Rs_{340 \times 1} = \sum_{j=1}^{3 \times N - 3} (r_{1,j})_{1 \times (3 \times N - 3)} \quad (8)$$

$$\oplus \sum_{j=1}^{3 \times N - 3} (r_{2,j})_{1 \times (3 \times N - 3)} \oplus \dots \oplus \sum_{j=1}^{3 \times N - 3} (r_{340,j})_{1 \times (3 \times N - 3)}$$

$$p_{i,h} = \begin{cases} 1, & P_h \dots P_{h+Kp-1} = Q(i) \\ 0, & P_h \dots P_{h+Kp-1} \neq Q(i) \end{cases} \quad (9)$$

$$r_{i,j} = \begin{cases} 1, & R_j \dots R_{j+Kj-1} = T(i) \\ 0, & R_j \dots R_{j+Kj-1} \neq T(i) \end{cases} \quad (10)$$

$$i = 1, 2, \dots, 8420; h = 1, 2, \dots, 3 \times M - 3; l = 1, 2, \dots, 340;$$

$$j = 1, 2, \dots, 3 \times N - 3; Kp = 1, 2, 3; Kj = 1, 2, 3, 4$$

where $Q(i) = [A, C, \dots, Y, AA, AC, \dots, YY, AAA, AAC, \dots, YYY]$, $T(i) = [A, G, C, U, AA, AG, \dots, UU, AAA, AAG, \dots, UUU, AAAA, AAAG, \dots, UUUU]$, similarly, a $340 \times k$ matrix Rs transformed with k RNA sequences from k sample pairs.

[27], which uses a dynamic programming technique based on finding the minimum free energy. lncRNA secondary structures are represented in ring and stem regions and formed as brackets and dots. Correspondingly, the protein secondary structures are predicted by SSpro [28] based on α -helix (H), β -sheet (E), and coil (C) conformation parameters.

Similar to the sequence encoding process, we calculated the 1-3-mer frequency of protein secondary structure and the 1-4-mer frequency of RNA secondary structure as a second

3) Physicochemical information encoding

stacking energy, entropy, free energy, enthalpy, etc). Using Pse-in-One tool, we got a protein physicochemical property vector with 22 elements and an lncRNA physicochemical property vector with 18 elements. At last, we splice the RNA and protein physicochemical information vectors and form one $(22 + 18) \times k$ matrix Phc_k from k sample pairs.

C. serial network and parallel framework design

In this study, we design a different network framework combining convolution, batch normalization, activation, max pooling, dropout, and fully connected layer, referred as to serial network CSFFR_SER. Its structure is shown in Fig. 2. In the feature encoding process, we achieve three feature matrices: sequence matrix Seq_k , structure matrix Str_k , and physicochemical matrix Phc_k . Each matrix is divided into k and fed into the network in a k sample order. Convolutional neural network (CNN) is employed to extract and form high-level representations from the input. The sequence, structure, and physicochemical information encoding vectors are analyzed continuously to form tandem feature embedding representations. Finally, feature vectors are flattened and input to full connection layers, and the softmax activation function is employed at the last layer to make binary predictions.

In previous studies, the deep-learning network was often network, and each network integrated multiple features [15, 16,

Fig. 2. The structure of serial framework CSFFR SER.

composed of 2 separate networks: RNA network and protein [21], which ignored the differences among features and

attenuated the association between protein and lncRNA. In this study, we construct a serial framework CSFFR_SER fusing three kinds of features, and each feature integrates lncRNA and protein characteristics. CSFFR_SER firstly trains on the original sequence features, then splices the high-level output *Oseq* with structure features for training. To ensure the splicing dimension correspondence, we use convolutional layers with kernel size 1. Afterward, CSFFR_SER splices the high-level output *Oseq_str* with physicochemical properties for further learning. At last, the vectors of the output *Oseq_str_pse* are flattened and served as the input of two fully connected layers with 128 and 96 neurons, the last fully connected layer with 2 neurons being regarded as a classifier for prediction.

The pooling layer of CNN filters the redundant features, however, multiple pooling layers may greatly reduce the original important information of features. To obtain a broader feature representation, we adopt a method of feature reuse [30]. Each layer obtains additional inputs from all preceding layers. That is, for a network with L layers, instead of just one input for each layer in traditional architectures, the i -th layer has i inputs consisting of the feature maps of all preceding convolutional blocks. In an L -layer CNN, each layer implements a non-linear transformation, which can be a composite function of operations such as Batch Normalization (BN) [31], rectified linear units (ReLU) [32], Pooling [33], or Convolution (Conv). We apply three consecutive operations: a Conv with kernel-size

of 3 and filter quantity of 24, followed by a BN and ReLU to compose each convolutional block, named as *Block*.

For a comparison with the serial strategy, we additionally construct a parallel framework involving three feature matrices: sequence matrix, structure matrix, and physicochemical matrix. This parallel framework is referred as CSFFR_PAR. Its structure is shown in Fig. 3. In framework CSFFR_PAR, three similar characteristic embedding parts use consistent networks to analyze the sequence, structure, and physicochemical property input vectors separately and form three high-level characteristic representations which are followed by splicing. Then, the vectors of the characteristic output are flattened and fed into three fully connected layers with 128, 96, and 2 neurons.

The two frameworks use the softmax activation function at their last layers to make binary predictions and use the back-propagation algorithm [34] to minimize the loss function of binary cross entropy. The optimization method Adam [35] is employed to train each framework. In each convolution layer, batch normalization [31] is employed to accelerate the convergence rate of learning, and ReLU [32] activation function is used to overcome the problem of gradient disappearance. In addition, in the fully connected layer, we employ the dropout layer with the rate of 0.2 to prevent the overfitting of the deep learning network.

D. Evaluation criteria

Fig. 3. The structure of parallel framework CSFFR_PAR.

In this study, we evaluate the performance of LPI-CSFFR and other methods by seven metrics, accuracy (ACC), sensitivity (SEN), specificity (SPE), precision (PRE), F1-score (F1), Matthews correlation coefficient (MCC), and area under the curve (AUC) of the receiver operating characteristic (ROC). The formulas of the first six measurements are as follows:

$$ACC = \frac{TP + TN}{TP + TN + FP + FN} \quad (11)$$

$$SEN = \frac{TP}{TP + FN} \quad (12)$$

$$SPE = \frac{TN}{TN + FP} \quad (13)$$

$$PRE = \frac{TP}{TP + FP} \quad (14)$$

$$F1 = \frac{2 \times PRE \times REC}{PRE + REC}, REC = \frac{TP}{TP + FN} \quad (15)$$

$$MCC = \frac{TP \times TN - FP \times FN}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}} \quad (16)$$

Where TP and TN mean the number of predicted positive and negative samples correctly, respectively; FP and FN denote the number of predicted positive and negative samples incorrectly, respectively. All these evaluation metrics were calculated by

five-fold Cross-Validation (CV) and there were no overlaps between training and testing data.

III. RESULTS AND DISCUSSION

A. Comparison with negative sample random selection strategy

First, we construct another RPI1460 dataset, named RPI1460_random, by randomly pairing lncRNA and protein directly. RPI1460_random dataset contains 1460 positive pairs and 1460 negative pairs. The results of LPI-CSFFR on the RPI1460 and RPI1460_random datasets are shown in Table I and Fig. 4, from which we can see that the accuracy of LPI-CSFFR on RPI1460 is 83%, which is 1.8% higher than that on RPI1460_random, and MCC on RPI1460 is 0.671, which is 0.034 higher than that on RPI1460_random. Correspondingly, training with RPI1460, the sensitivity, specificity, precision, F1, AUC, are increased by 0.7%, 2.9%, 2%, 1.4%, 2.4% respectively compared with RPI1460_random. Values in bold represent the maximum number during the comparison. These results indicate that the strategy in this study to construct the negative samples can improve LPI-CSFFR prediction performance.

Table I
RESULTS OF LPI-CSFFR ON RPI1460 AND RPI1460_RANDOM DATASETS

Dataset	ACC (%)	SEN (%)	SPE (%)	PRE (%)	F1 (%)	MCC	AUC (%)
RPI1460	83.1±1.1	91±1.2	75.2±1.1	78.6±0.9	84.3±1.1	0.671±0.023	86.6±1
RPI1460_random	81.3±0.9	90.3±0.8	72.3±2.4	76.6±1.4	82.9±0.6	0.637±0.015	84.2±1.1

Fig.4 Performance comparison between RPI1460 and RPI1460_random dataset.

B. Comparison with the parallel framework

We test two different frameworks in this study including serial framework CSFFR_SER and parallel framework

CSFFR_PAR. Fig. 5 shows the ROC curves generated on the RPI1460 and RPI1807 datasets. Table II lists the prediction results on the RPI1807 and RPI1460 dataset, from which we can see that in RPI1807, when using CSFFR_SER, the accuracy is 98%, the sensitivity is 98.2%, the specificity is 97.8%, the precision is 97.8%, the F1 is 97.9%, the MCC is 0.96, and the AUC 99.5%, which are 1.6%, 1.2%, 2%, 2.1%, 1.5%, 0.031, and 2.9% higher than that on CSFFR_PAR respectively. This gap is even more pronounced on the RPI1460, in which the accuracy based on CSFFR_PAR is 62.3%, 20.8% lower than based on CSFFR_SER. Similarly, adopting CSFFR_PAR the sensitivity is 24.7%, the specificity 17%, the precision 22.4%, the F1 25.9%, the MCC 0.415, the AUC 22.5% lower than using CSFFR_SER. The results show the effectiveness of the tandem framework. Taking the same framework, the training on the RPI1807 dataset is much better than the RPI1460, which also shows the superiority of building negative samples based on the threshold.

C. Comparison with other methods

To assess the performance of our LPI-CSFFR, we compared LPI-CSFFR with other state-of-the-art methods, including RPISeq-RF, IPMiner, CFRP, and RPITER on RPI1460 and RPI1807 datasets. The results of LPI-CSFFR and the other four methods are shown in Table III. From the results on RPI1460, we can see that the accuracy, sensitivity, F1, MCC value of LPI-CSFFR are the highest among the compared five methods, which are 4.4%, 12.6%, 5.6%, and 0.096 higher than those of RPISeq-RF, and 7.6%, 7.5%, 7%, and 0.152 higher than those of IPMiner, and 1.9%, 12.1%, 2.7%, and 0.046 higher than those of RPITER. The specificity, precision, and AUC of LPI-CSFFR are 7.8%, 6.2%, and 11.1% higher than those of IPMiner. In addition, the precision of LPI-CSFFR is 33.1% higher compared with RPITER, and the AUC is 8.1 higher compared with RPISeq-RF. When training on the RPI1807 dataset, LPI-CSFFR obtains the highest sensitivity values, which shows the high recognition of the positive samples, 1.1% higher than that of RPISeq-RF, 1.3% higher than that of IPMiner, and 0.4% higher than that of CFRP, and 12.3% higher than that of RPITER. In the meantime, as compared to RPITER, the accuracy of LPI-CSFFR is 7.7% higher, the specificity 3.3% higher, the precision 13.4%, the F1 13.2% higher, and the MCC 0.156 higher. At the AUC, the LPI-CSFFR is 1.5% higher than RPISeq-RF, and 1.9% higher than IPMiner. Besides, we can see that CFRP and RPISeq-RF which both applied the RF classifier perform well on the RPI1807, and CFRP extra acting a nonlinear transformation of the original

features seems better, which show that based on good quality data with feature vectors carefully designed may well improve the model performance. LPI-CSFFR performs as well as CFRP excellently on the RPI1807 that the differences in accuracy, specificity, precision, F1, MCC, and AUC are not more than 0.8%. These results indicate that our LPI-CSFFR performs well in predicting lncRNA-protein interactions.

D. Performance on an independent dataset

To further validate the ability of LPI-CSFFR to predict the interaction among lncRNA and protein, we try to use the RPI1460 dataset to train the deep learning model and verify it on the RPI9373 dataset. There is no overlap between the two datasets. RPI9373 can be divided into 6 organisms, and we conduct experiments on them separately. As shown in table IV, our method predicts the correct number of pairs of interactions on Homo sapiens (*H. sapiens*), Caenorhabditis Elegans (*C. elegans*), Mus musculus (*M. musculus*), Drosophila melanogaster (*D. melanogaster*), Saccharomyces cerevisiae (*S. cerevisiae*) and Escherichia coli (*E. coli*) for 6628, 4, 1988, 54, 427 and 72, with an accuracy rate of 97.9%, 100%, 98.2%, 94.7%, 96.6%, and 92.3%, respectively. On the independent dataset RPI9373, we finally predicted the correct number of RNA-protein pairs to be 9173, with a total accuracy of up to 97.9%.

E. Application of LPI-CSFFR to establish lncRNA-protein network

We further use the trained model from RPI1460 on RPI9373 dataset to build the lncRNA-protein networks. By making the use of the tool Cytoscape [36] for network clustering, the nodes in the network are lncRNAs and proteins, and the edges are the relationship between them. For *M. musculus* in the RPI9373, we correctly predict 1988 of 2024 interactions. Among them, 601 pairs from the hot-spot protein Q4VA08, 1033 pairs from the protein Q05C33 are 100% correctly predicted. Due to the too large number of interaction pairs, we construct the lncRNA-protein networks according to the prediction results that exclude proteins Q4VA08 and Q05C33. The lncRNA-protein networks we build are shown in Fig. 6. From the result, we find the hot spot proteins Q4VA08, Q05C33, Q00899, O09106, Q8CHK4, Q80U70, Q80Y84, Q7TPM0, P84104, Q61188, Q9CQJ4, O88974, O54864, P41230, and I6L9I7. Building the interaction network helps us to visualize how many hot spot proteins have been identified, and helps to explore new potential RNA and protein interactions.

Table II
RESULTS OF CSFFR_SER AND CSFFR_PAR ON RPI1460 AND RPI1807 DATASETS

Dataset	Framework	ACC (%)	SEN (%)	SPE (%)	PRE (%)	F1 (%)	MCC	AUC (%)
RPI	CSFFR_SER	83.1±1.1	91±1.2	75.2±1.1	78.6±0.9	84.3±1.1	0.671±0.023	86.6±1
1460	CSFFR_PAR	62.3±3.3	66.3±12.2	58.2±5.8	56.2±8.3	58.4±10	0.256±0.068	63.9±4.5
RPI	CSFFR_SER	98±0.4	98.2±1	97.8±1.1	97.8±1.1	97.9±0.4	0.96±0.007	99.5±0.3
1807	CSFFR_PAR	96.4±0.8	97±0.7	95.8±1.8	95.7±1.7	96.4±0.7	0.929±0.015	96.6±0.7

Fig.5 Performance comparison between CSFFR_SER and CSFFR_PAR.
(a) Performance on RPI1460 dataset. (b) Performance on RPI1807 dataset.

Table III
RESULTS OF LPI-CSFFR AND OTHER FOUR METHODS
ON RPI1460 AND RPI1807 DATASETS IN 5CV TEST

Dataset	Method	ACC (%)	SEN (%)	SPE (%)	PRE (%)	F1 (%)	MCC	AUC (%)
RPI1460	LPI-CSFFR	83.1±1.1	91±1.2	75.2±1.1	78.6±0.9	84.3±1.1	0.671±0.023	86.6±1
	RPISeq-RF	78.7±1.8	78.4±1.8	79.1±2.4	79±2.1	78.7±1.8	0.575±0.036	78.7±1.8
	IPMIner	75.5±2.2	83.5±4.7	67.4±2.7	72.4±1.5	77.3±2.5	0.519±0.044	75.5±2.2
	CFRP	81.2±0.6	78.9±0.9	79.8±0.4	83.5±1.9	81.6±0.8	0.625±0.013	87.5±0.4
	RPITER	69.5±2.4	48.8±6	90.2±2.1	61±10.6	51.2±6.1	0.412±0.053	90.4±0.8
RPI1807	LPI-CSFFR	98±0.4	98.2±1	97.8±1.1	97.8±1.1	97.9±0.4	0.96±0.007	99.5±0.3
	RPISeq-RF	98±0.6	97.1±0.7	98.8±0.7	98.7±0.8	97.9±0.6	0.96±0.012	98±0.6
	IPMIner	97.5±0.5	96.9±0.5	98.2±1.2	98.1±1.2	97.5±0.5	0.951±0.009	97.6±0.4
	CFRP	98.2±0.2	97.8±0.2	98.6±0.1	98.5±0.1	98.1±0.2	0.964±0.003	99.7±0.1
	RPITER	90.3±7.6	85.9±17.4	94.5±8.7	84.4±16	84.7±16.4	0.804±0.156	99.7±0.4

TABLE IV
THE PREDICTED PERFORMANCE OF TRAINED MODEL FROM RPI1460 ON RPI9373

organism	The number of pairs of RNA-protein interactions	The correctly predicted number of pairs of RNA-protein interactions	ACC (%)
H. sapiens	6768	6628	97.9
C. elegans	4	4	100
M. musculus	2024	1988	98.2
D. melanogaster	57	54	94.7
S. cerevisiae	442	427	96.6
E. coli	78	72	92.3
Total	9373	9173	97.9

Fig. 6. The *M. musculus* networks were constructed based on interaction pairs predicted by LPI-CSFFR. The octagon green and the circular yellow nodes represent the protein and RNA, respectively. The black and red edges indicate the correct and the incorrect prediction of the lncRNA-protein interactions, respectively.

Fig. 6. (Continued) The *M. musculus* networks were constructed based on interaction pairs predicted by LPI-CSFFR. The octagon green and the circular yellow nodes represent the protein and RNA, respectively. The black and red edges indicate the correct and the incorrect prediction of the lncRNA-protein interactions, respectively.

Fig. 6. (Continued) The *M. musculus* networks were constructed based on interaction pairs predicted by LPI-CSFFR. The octagon green and the circular yellow nodes represent the protein and RNA, respectively. The black and red edges indicate the correct and the incorrect prediction of the lncRNA-protein interactions, respectively.

Fig. 6. (Continued) The *M. musculus* networks were constructed based on interaction pairs predicted by LPI-CSFFR. The octagon green and the circular yellow nodes represent the protein and RNA, respectively. The black and red edges indicate the correct and the incorrect prediction of the lncRNA-protein interactions, respectively.

F. Case study

In order to assess the predictive ability of LPI-CSFFR, we collected new wet experiment data from NPInter v4.0 in the *H. sapiens* and obtained 93 brand-new positive pairs including 10 proteins and 89 lncRNAs. NPInter v4.0 is an upgraded version of NPInter v2.0 proposed by Teng *et al.*, which includes the experimentally validated interactions between RNA and proteins [37]. Specifically, we paired all the protein and lncRNA pairs in the order of protein and lncRNA id, and formed 890 sample pairs including 93 pairs confirmed by experiment and 797 pairs unconfirmed. Then we use the model trained with RPI1460 dataset and predict the 890 samples. The prediction results show that all experimentally validated positive pairs have been accurately predicted. Table V lists the top 10 RNA-protein pairs of the predicted scores and the comparison results with NPInter v4.0. From the table, we can see that 6 out of the top 10 can be verified. We noted that pairs that are not confirmed by experiments may also be interactive pairs. The results of this experiment show that LPI-CSFFR is promising to discover potential RNA-protein interactions.

TABLE V
TOP 10 RNA-PROTEIN PAIRS WERE PREDICTED
BY LPI-CSFFR BASED ON KNOWN INTERACTIONS

RNA	Protein	Evidence
NONHSAG041027	ELAVL1	confirmed
NONHSAG003899	ELAVL1	confirmed
NONHSAG053895	ELAVL1	confirmed
NONHSAG053897	ELAVL1	confirmed
NONHSAG045208	ELAVL1	confirmed
NONHSAG047292	ELAVL1	confirmed
NONHSAG048098	CPSF7	unconfirmed
NONHSAG010565	CPSF7	unconfirmed
NONHSAG044905	CPSF7	unconfirmed
NONHSAG041027	CPSF7	unconfirmed

IV. CONCLUSION

In this study, we adopt a relatively reliable negative sample construction strategy based on sequence similarity to reduce the probability of choosing false-negative samples as negative pairs. Then we fuse the raw sequence information, secondary structure information, and physicochemical properties of protein and RNA in a tandem fusion manner to dig out the

potential connection between the individual features during the subsequent training session. We further propose a novel CNN-based method by a feature reuse way that maximizes the contribution degree of each feature to predict whether specific lncRNA-protein pairs interact, named LPI-CSFFR. The experimental results demonstrate that the performance of our LPI-CSFFR is better than other state-of-the-art methods in predicting lncRNA-protein interactions. We use the serial framework called CSFFR_SER to fuse the lncRNA and protein interaction related information, which makes the use of the dependencies in three different features, further enhancing the predictive performance of the model. To verify the reliability and effectiveness of our LPI-CNNCP in predicting lncRNA-protein interactions, we compare it with the other four methods of IPMiner, RPISeq-RF, RRPITER, and CFRP in the 5-fold cross validation test. LPI-CSFFR achieves the best performance, indicating that our LPI-CSFFR is promising. In addition, we test the LPI-CSFFR model trained with RPI1460 dataset on an independent RPI9373 dataset, and draw the RPI networks on the basis of this result. These good experimental results show that the proposed method is reliable for predicting lncRPIs. In addition, the model is able to be used to predict new potentially positive pairs. In the future work, we will consider improving the method of feature encoding to achieve higher prediction accuracy and exploring the interpretability of the model. In terms of application, such serial fusion with feature reuse method based on CNN can be applied for the prediction of other biological processes, such as protein-protein interaction, ncRNA-disease-related predictions, etc. The source code and the datasets used in this study are available at <https://github.com/JianjunTan-Beijing/LPI-CSFFR>.

COMPETING INTERESTS

The authors declare that they have no competing interests.

CONFLICTS OF INTEREST

The authors declare that there is no conflict of interest regarding the publication of this paper.

REFERENCES:

- [1] M.U. Kaikkonen, M.T. Lam and C.K. Glass, "Non-coding RNAs as regulators of gene expression and epigenetics," *Cardiovasc. Res.*, vol. 90, no. 3, pp. 430-440, 2011, doi: 10.1093/cvr/cvr097.
- [2] D. Umlauf, P. Fraser and T. Nagano, "The role of long non-coding RNAs in chromatin structure and gene regulation: variations on a theme," *Biol. Chem.*, vol. 389, no. 4, pp. 323-331, 2008, doi: 10.1515/BC.2008.047.
- [3] X. Wang, *et al.*, "Induced ncRNAs allosterically modify RNA-binding proteins in cis to inhibit transcription," *Nature*, vol. 454, no. 7200, pp. 126-130, 2008, doi: 10.1038/nature06992.
- [4] Y. Ogawa, B.K. Sun and J.T. Lee, "Intersection of the RNA interference and X-inactivation pathways," *Science*, vol. 320, no. 5881, pp. 1336-1341, 2008, doi: 10.1126/science.1157676.
- [5] E. Schmidt, *et al.*, "LincRNA H19 protects from dietary obesity by constraining expression of monoallelic genes in brown fat," *Nat. Commun.*, vol. 9, no. 1, pp. 3622, 2018, doi: 10.1038/s41467-018-05933-8.
- [6] S. Schoeftner, *et al.*, "Recruitment of PRC1 function at the initiation of X inactivation independent of PRC2 and silencing," *Embo J.*, vol. 25, no. 13, pp. 3110-3122, 2006, doi: 10.1038/sj.emboj.7601187.
- [7] J.L. Rinn, *et al.*, "Functional Demarcation of Active and Silent Chromatin Domains in Human HOX Loci by Noncoding RNAs," *Cell*, vol. 129, no. 7, pp. 1311-1323, 2007, doi: 10.1016/j.cell.2007.05.022.
- [8] E. Salta and B. De Strooper, "Noncoding RNAs in neurodegeneration," *Nat. Rev. Neurosci.*, vol. 18, no. 10, pp. 627-640, 2017, doi: 10.1038/nrn.2017.90.
- [9] L. Sun and J.D. Lin, "Function and Mechanism of Long Noncoding RNAs in Adipocyte Biology," *Diabetes*, vol. 68, no. 5, pp. 887-896, 2019, doi: 10.2337/dbi18-0009.
- [10] R.A. Gupta, *et al.*, "Long non-coding RNA HOTAIR reprograms chromatin state to promote cancer metastasis," *Nature*, vol. 464, no. 7291, pp. 1071-1076, 2010, doi: 10.1038/nature08975.
- [11] D. Marchese, N.S. de Groot, N. Lorenzo Gotor, C.M. Livi and G.G. Tartaglia, "Advances in the characterization of RNA-binding proteins," *Wiley Interdisciplinary Reviews: RNA*, vol. 7, no. 6, pp. 793-810, 2016, doi: 10.1002/wrna.1378.
- [12] M. Ramanathan, D.F. Porter and P.A. Khavari, "Methods to study RNA-protein interactions," *Nat. Methods*, vol. 16, no. 3, pp. 225-234, 2019, doi: 10.1038/s41592-019-0330-1.
- [13] X. Pan, Y. Yang, C.Q. Xia, A.H. Mirza and H.B. Shen, "Recent methodology progress of deep learning for RNA-protein interaction prediction," *Wiley Interdisciplinary Reviews: RNA*, vol. 10, no. 6, pp. e1544, 2019, doi: 10.1002/wrna.1544.
- [14] V. Suresh, L. Liu, D. Adjero and X. Zhou, "RPI-Pred: predicting ncRNA-protein interaction using sequence and structural information," *Nucleic Acids Res.*, vol. 43, no. 3, pp. 1370-1379, 2015, doi: 10.1093/nar/gkv020.
- [15] L. Wang, X. Yan, M. Liu, K. Song, X. Sun and W. Pan, "Prediction of RNA-protein interactions by combining deep convolutional neural network with feature selection ensemble method," *J. Theor. Biol.*, vol. 461, pp. 230-238, 2019, doi: 10.1016/j.jtbi.2018.10.029.
- [16] C. Peng, S. Han, H. Zhang and Y. Li, "RPITER: A Hierarchical Deep Learning Framework for ncRNA-Protein Interaction Prediction," *Int. J. Mol. Sci.*, vol. 20, no. 5, pp. 1070, 2019, doi: 10.3390/ijms20051070.
- [17] U.K. Muppurala, V.G. Honavar and D. Dobbs, "Predicting RNA-protein interactions using only sequence information," *BMC Bioinformatics*, vol. 12, pp. 489, 2011, doi: 10.1186/1471-2105-12-489.
- [18] Q. Lu, *et al.*, "Computational prediction of associations between long non-coding RNAs and proteins," *BMC Genomics*, vol. 14, pp. 651, 2013, doi: 10.1186/1471-2164-14-651.
- [19] X. Pan, Y. Fan, J. Yan and H. Shen, "IPMiner: hidden ncRNA-protein interaction sequential pattern mining with stacked autoencoder for accurate computational prediction," *BMC Genomics*, vol. 17, pp. 582, 2016, doi: 10.1186/s12864-016-2931-8.
- [20] Q. Dai, M. Guo, X. Duan, Z. Teng and Y. Fu, "Construction of Complex Features for Computational Predicting ncRNA-Protein Interaction," *Frontiers in Genetics*, vol. 10, 2019, doi: 10.3389/fgene.2019.00018.
- [21] J. Wang, *et al.*, "EDLMFC: an ensemble deep learning framework with multi-scale features combination for ncRNA-protein interaction prediction," *BMC Bioinformatics*, vol. 22, no. 1, pp. 133, 2021, doi: 10.1186/s12859-021-04069-9.
- [22] S.K. Burley, *et al.*, "RCSB Protein Data Bank: powerful new tools for exploring 3D structures of biological macromolecules for basic and applied research and education in fundamental biology, biomedicine, biotechnology, bioengineering and energy sciences," *Nucleic Acids Res.*, vol. 49, no. D1, pp. D437-D451, 2021, doi: 10.1093/nar/gkaa1038.
- [23] J. Yuan, W. Wu, C. Xie, G. Zhao, Y. Zhao and R. Chen, "NPInter v2.0: an updated database of ncRNA interactions," *Nucleic Acids Res.*, vol. 42, no. Database issue, pp. D104-D108, 2014, doi: 10.1093/nar/gkt1057.
- [24] Z. Cheng, K. Huang, Y. Wang, H. Liu, J. Guan and S. Zhou, "Selecting high-quality negative samples for effectively predicting protein-RNA interactions," *BMC Syst. Biol.*, vol. 11, no. Suppl 2, pp. 9, 2017, doi: 10.1186/s12918-017-0390-8.
- [25] T.F. SMITH and M.S. WATERMAN, "IDENTIFICATION OF COMMON MOLECULAR SUBSEQUENCES," *J. Mol. Biol.*, vol. 147, no. 1, pp. 195-197, 1981, doi: 10.1016/0022-2836(81)90087-5.
- [26] B. Liu, H. Wu and K. Chou, "Pse-in-One 2.0: An Improved Package of Web Servers for Generating Various Modes of Pseudo Components of DNA, RNA, and Protein Sequences," *Natural Science*, vol. 09, no. 04, pp. 67-91, 2017, doi: 10.4236/ns.2017.94007.
- [27] R. Lorenz, *et al.*, "ViennaRNA Package 2.0," *Algorithm. Mol. Biol.*, vol. 6, 2011, doi: 10.1186/1748-7188-6-26.
- [28] C.N. Magnan and P. Baldi, "SSpro/ACCpro 5: almost perfect prediction of protein secondary structure and relative solvent accessibility using profiles, machine learning and structural similarity," *Bioinformatics*, vol. 30, no. 18, pp. 2592-2597, 2014, doi: 10.1093/bioinformatics/btu352.
- [29] S. Kawashima, P. Pokarowski, M. Pokarowska, A. Kolinski, T. Katayama and M. Kanehisa, "AAindex: amino acid index database, progress report 2008," *Nucleic Acids Res.*, vol. 36, no. Database, pp. D202-D205, 2007, doi: 10.1093/nar/gkm998.
- [30] G. Huang, Z. Liu, L. van der Maaten, K.Q. Weinberger and IEEE, *Densely Connected Convolutional Networks*, Book

Densely Connected Convolutional Networks, Series Densely Connected Convolutional Networks, ed., Editor ed., 2017, pp. 2261-2269.

[31] S. Ioffe and C. Szegedy, Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift, Book Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift, Series Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift 37, ed., Editor ed., 2015, pp. 448-456.

[32] X. Glorot A. Bordes and Y. Bengio, Deep sparse rectifier neural networks, Microtome Publishing, 2011, pp. 315-323.

[33] Y. Lecun, L. Bottou, Y. Bengio and P. Haffner, "Gradient-based learning applied to document recognition," P. Ieee, vol. 86, no. 11, pp. 2278-2324, 1998, doi: 10.1109/5.726791.

[34] D.E. RUMELHART G.E. HINTON and R.J. WILLIAMS, "LEARNING REPRESENTATIONS BY BACK-PROPAGATING ERRORS," Nature, vol. 323, no. 6088, pp. 533-536, 1986, doi: 10.1038/323533a0.

[35] D.P. Kingma and J.L. Ba, Adam: A method for stochastic optimization, International Conference on Learning Representations, ICLR, 2015.

[36] D. Otasek, J.H. Morris, J. Boucas, A.R. Pico and B. Demchak, "Cytoscape Automation: empowering workflow-based network analysis," GENOME BIOLOGY, vol. 20, no. 1, 2019, doi: 10.1186/s13059-019-1758-4.

[37] X. Teng, *et al.*, "NPInter v4.0: an integrated database of ncRNA interactions," Nucleic Acids Res., vol. 48, no. D1, pp. D160-D165, 2020, doi: 10.1093/nar/gkz969.