Journal of Theoretical Biology 461 (2019) 230–238

Contents lists available at ScienceDirect

Journal of Theoretical Biology

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

Prediction of RNA-protein interactions by combining deep convolutional neural network with feature selection ensemble method

Lei Wang[a,*], Xin Yan[b,**], Meng-Lin Liu[a], Ke-Jian Song[c], Xiao-Fei Sun[a], Wen-Wen Pan[a]

[a] College of Information Science and Engineering, Zaozhuang University, Zaozhuang, Shandong 277100, China
[b] School of Foreign Languages, Zaozhuang University, Zaozhuang, Shandong 277100, China
[c] School of Information Engineering, JiangXi University of Science and Technology, Ganzhou, Jiangxi 341000, China

ARTICLE INFO

Article history:
Received 1 December 2017
Revised 22 March 2018
Accepted 11 October 2018
Available online 12 October 2018

Keyword:
RNA-protein interaction
Convolution neural network
Extreme learning machine
Position-specific scoring matrix

ABSTRACT

RNA-protein interaction (RPI) plays an important role in the basic cellular processes of organisms. Unfortunately, due to time and cost constraints, it is difficult for biological experiments to determine the relationship between RNA and protein to a large extent. So there is an urgent need for reliable computational methods to quickly and accurately predict RNA-protein interaction. In this study, we propose a novel computational method RPIFSE (predicting RPI with Feature Selection Ensemble method) based on RNA and protein sequence information to predict RPI. Firstly, RPIFSE disturbs the features extracted by the convolution neural network (CNN) and generates multiple data sets according to the weight of the feature, and then use extreme learning machine (ELM) classifier to classify these data sets. Finally, the results of each classifier are combined, and the highest score is chosen as the final prediction result by weighting voting method. In 5-fold cross-validation experiments, RPIFSE achieved 91.87%, 89.74%, 97.76% and 98.98% accuracy on RPI369, RPI2241, RPI488 and RPI1807 data sets, respectively. To further evaluate the performance of RPIFSE, we compare it with the state-of-the-art support vector machine (SVM) classifier and other exiting methods on those data sets. Furthermore, we also predicted the RPI on the independent data set NPInter2.0 and drew the network graph based on the prediction results. These promising comparison results demonstrated the effectiveness of RPIFSE and indicated that RPIFSE could be a useful tool for predicting RPI.

© 2018 Elsevier Ltd. All rights reserved.

1. Introduction

The interaction among protein and ribonucleic acid is closely related to the most basic life activities of organisms (Chen and Varani, 2005; Cooper et al., 2011; Lukong et al., 2008). Many key cellular processes such as signal transduction, chromosome replication, substance transport, mitosis, transcription and translation are inseparable from RNA-protein interaction (Coelho et al., 2017; Zhang et al., 2016). Therefore, exploring the molecular mechanism of RPI is of important guiding significance for understanding biological processes, pathology research and drug design.

At present, biologists have less structural data of protein complexes obtained by X-ray diffraction, nuclear magnetic resonance, electron microscope and neutron diffraction. This is mainly because the experimental methods have the disadvantage of complex measurement process, time-consuming and cost-intensive (Ke and Doudna, 2004; Pai et al., 2017; Scott and Hennig, 2008). Especially with the development of high-throughput sequencing technology, people can quickly get a lot of transcriptome and proteome information, which contains a large number of potential RPI needs to analysis. However, the traditional experimental method can only be studied at a given protein, RNA or protein-RNA complex, technically far from meeting this demand, so there is an urgent need to develop the reliable computational approaches to accurately predict RPI (Guo et al., 2008).

Researchers have made many achievements in the prediction of RPI using computational methods (Anbarasu and Sethumadhavan, 2007; Fujishima et al., 2007; Kumar et al., 2011; Yu et al., 2006). For instance, Muppirala et al. constructed the RPISeq model using only sequence information. The model first converts the protein sequence into a sequence of physical and chemical properties containing seven states, then calculates the 3-mer composition of the protein physical and chemical sequence, and calculated the 4-mer composition of the RNA sequence directly. When using SVM and random forest classifier to predict, the model obtained the AUC values as high as 0.96 and 0.92, respectively (Muppirala et al.,

* Corresponding author.
** Co-corresponding author.
E-mail addresses: leiwang@ms.xjb.ac.cn (L. Wang), xinyanuzz@gmail.com (X. Yan), kjsong@aliyun.com (K.-J. Song), sxf@uzz.edu.cn (X.-F. Sun), panwenwen@uzz.edu.cn (W.-W. Pan).

https://doi.org/10.1016/j.jtbi.2018.10.029
0022-5193/© 2018 Elsevier Ltd. All rights reserved.

2011). Bellucci et al. proposed the catRAPID method for large-scale computation of RPI. This method describes the corresponding proteins and RNA sequences by calculating the secondary structure information, hydrogen bonding and van der Waals forces of each protein and RNA molecules. Eventually, the catRAPID method correctly predicted 89% of the experimentally supported interactions (Bellucci et al., 2011). Lu et al. extracted the features of protein and RNA respectively, calculated the eigenvector by Fourier transform and matrix multiplication, and obtained the correlation coefficient between protein and RNA interaction, so as to accurately predicted RPI (Lu et al., 2013).

In this paper, we propose a new model RPIFSE to predict RNA-protein interaction using feature selection ensemble method. For a given pair of RNA-protein sequences, RPIFSE can predict whether there is an interaction between them. Specifically, we first convert RNA and protein sequences into numerical matrices that can be processed by computers, and then use the convolutional neural network (CNN) to extract their features. In order to improve the performance of the ensemble algorithm, we consider the perturbation of the feature space of the sample. We use the χ^2 statistical algorithm to sort the weight of the feature, and then select the features according to the given proportions. In addition, we also selected whether or not CNN was fine-tuned when extracting features. Through the combination of these methods, we generate a variety of samples and use them to train extreme learning machine (ELM) basic classifiers. In the integration operation, we follow the idea of the attention mechanism to calculate the weight of the basic classifier and thus the final prediction results. In the experiment, we evaluate the performance of RPIFSE on different data sets and compare it with other state-of-the-art methods. Excellent results show that RPIFSE has not only high prediction accuracy, but also has strong generalization ability and robustness.

2. Materials and methods

2.1. Data sets

In order to test the performance and applicability of RPIFSE, we validate it on different RNA-protein interaction data sets, including RPI369 (Muppirala et al., 2011), RPI2241 (Muppirala et al., 2011), RPI488 (Pan et al., 2016), RPI1807 (Suresh et al., 2015) and NPInter2.0 (Yuan et al., 2014). These data sets contain mRNA-protein and ncRNA-protein data, respectively. Taking the RPI1807 data set as an example, it contains 1807 positive RNA-protein pairs, which are composed of 1807 protein chains and 1078 RNA chains, and 1436 negative RNA-protein pairs, which are composed of 1436 protein chains and 493 RNA chains. These data are collected separately in the protein-RNA interface database (PRIDB) (Lewis et al., 2011) and nucleic acid database (NDB) (Lu et al., 2013). According to the method of study (Suresh et al., 2015), we first use the EMBOSS needle program (Rice et al., 2000) to remove sequence similarity of more than 30% of the RNA and protein chain. Then we further tested the atomic interactions with a distance threshold (3.4Å) among these non-redundant pairs. According to the research of Rajagopal and Vishveshwara (2005), the threshold 3.4 Å is reasonable and sufficient to cover 'strong' and 'moderate' hydrogen bonds and energy-rich van der Waals contacts. So we set the threshold to distinguish the strongly interacting protein-RNA pairs and weakly interacting protein-RNA pairs. Finally delete the RNA sequence of less than 15 nucleotides in length and the protein sequence of less than 25 amino acids in length. In this way, the final RPI1807 data set is constructed. RPI369, RPI2241, RPI488 and NPInter2.0 data sets are also constructed according to the corresponding methods, where RPI369, RPI2241 and RPI488 are extracted from the structural-based experimental complex, and NPInter2.0 is obtained from the physical association among the proteins and the ncRNAs. The number of RNA-protein pairs with interaction contained in these data sets is 369, 2241, 488 and 10,412, respectively. It should be noted that in the NPInter2.0 data set contains only RNA-protein pairs with interaction, so we use N/A to represent the number of RNA-protein pairs with non-interaction. The details of those data sets are summarized in Table 1.

2.2. The numerical descriptor representation of the sequence

In order to fully express the information hidden in the sequence and facilitate computer processing, we use different numerical transformation methods to deal with different sequences. For the RNA sequences, we use the order-preserving transformation (OPT) (Nashimoto, 2001; Yu and Huang, 2012) algorithm, and for the protein sequence, we use the position-specific scoring matrix (PSSM) (Cheng-Wei et al., 2008; Gribskov et al., 1987; Wang et al., 2017b; Xu et al., 2015) algorithm to transform them into numerical matrices.

The OPT algorithm takes full account of the location information of adjacent nucleotides in the design process and transforms each RNA character sequence into a numerical sparse matrix. Considering an RNA sequence $S =' s_1 s_2 \cdots s_L'$, where $s_i \epsilon \{A, C, G, U\}$, $i = 1, 2, \cdots, L$ and L means the length of the RNA sequence. We set up three consecutive letters as a 3-tuple, and use it to scan the sequence of RNA letters, thus forming a numerical sparse matrix. In particular, we sequentially scan three adjacent RNA alphabet sequences and form a new sequence as shown below, $(1, 2, 3), (2, 3, 4), \cdots, (L-2, L-1, L)$, and the length of which is $L-2$. Since the RNA sequence consists of 4 characters (A, C, G, U), the number of 3-tuple we construct is $4 \times 4 \times 4 = 64$. Finally, we construct an $64 \times (L-2)$ numerical sparse matrix M which can be represented as:

$$M = \left(m_{i,j}\right)_{64 \times (L-2)}, \ m_{i,j} = \begin{cases} 1, & s_j s_{j+1} s_{j+2} = T(i) \\ 0, & others \end{cases} \quad i = 1, 2, \cdots, 64; \ \ j = 1, 2, \cdots, L-2 \tag{1}$$

where $T(i) = [AAA, AAC, \cdots, UUU]$. Through the OPT algorithm we convert the RNA character sequence to the numerical sparse matrix.

We use the PSSM algorithm that can represent the biological evolutionary information to do the numerical matrix transformation of the protein sequence (Gao et al., 2016; Wang et al., 2016; Wang et al., 2017a; Zahiri et al., 2013). The PSSM algorithm was introduced at the beginning to detect distantly related protein and later achieved significant results in protein binding site prediction (Chen and Jeong, 2009), protein secondary structure prediction (Jones, 1999), and prediction of disordered regions (Jones and Ward, 2003). A complete PSSM matrix is composed of L rows and 20 columns, where L corresponds to the number of characters in the protein sequence and 20 represents the number of naive amino acids. Assuming PSSM $= \{\mathfrak{p}_{i,j}, \ i = 1 \cdots L \ and \ j = 1 \cdots 20\}$, the structure of PSSM can be expressed as:

$$PSSM = \begin{bmatrix} \mathfrak{p}_{1,1} & \mathfrak{p}_{1,2} & \cdots & \mathfrak{p}_{1,20} \\ \mathfrak{p}_{2,1} & \mathfrak{p}_{2,2} & \cdots & \mathfrak{p}_{2,20} \\ \vdots & \vdots & \vdots & \vdots \\ \mathfrak{p}_{L,1} & \mathfrak{p}_{L,2} & \cdots & \mathfrak{p}_{L,20} \end{bmatrix} \tag{2}$$

where $\mathfrak{p}_{i,j}$ in the i row of PSSM mean that the probability of the ith residue being mutated into type j of 20 native amino acids during the procession of evolutionary in the protein from multiple sequence alignments. In practice, we use the well-known Position-Specific Iterated BLAST (PSI-BLAST) (Altschul et al., 1997) toolkit to implement the algorithm. In order to obtain better experimental

Table 1
The details of the collected data sets.

Data set	Number of RNAs	Number of proteins	Number of interaction pairs	Number of non-interaction pairs
RPI369	332	338	369	369
RPI2241	842	2043	2241	2241
RPI488	25	247	243	245
RPI1807	1078	1807	1807	1436
NPInter2.0	4636	449	10,412	N/A

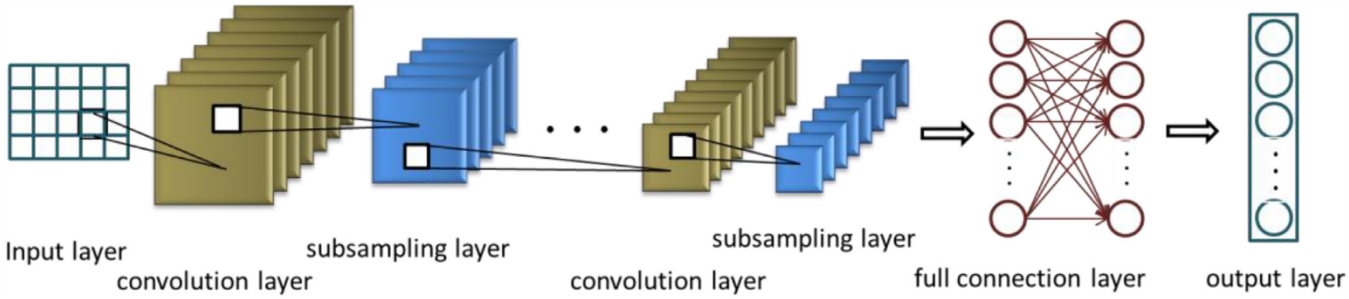

Fig. 1. Schematic diagram of CNN structure.

results, we set the PSI-BLAST value of e-value to 0.001, the number of iterations to 3, and compare with the *SwissProt* database. PSI-BLAST and *SwissProt* database applications can be downloaded at http://blast.ncbi.nlm.nih.gov/Blast.cgi.

2.3. Convolutional neural network

Convolutional neural network (CNN) is an efficiency identification algorithm which has been developed in recent years and has attracted extensive attention (Kalchbrenner et al., 2014). It introduces the thought of deep learning into the neural network, and extracts the features of different levels of the raw samples by convolution operation, thus producing the most suitable classification features. Therefore, we apply CNN to RNA-protein interaction prediction in the hope of extracting representative feature information.

The structure of the convolution neural network is shown in Fig. 1. It can be seen from the figure, CNN is a multi-layer neural network consisting of input layer, convolution layer, subsampling layer, full connection layer and the output layer. The mapping process is forward propagation and the output of the previous layer as the input of the current layer. Suppose that P_i is a feature graph of the *ith* layer, and it can be described as:

$$P_i = f(P_{i-1} \bullet W_i + b_i) \tag{3}$$

where $f(x)$ denotes the activation function, operator $\bullet$ represents convolution operations, W_i is the weight matrix of the convolution kernel of *ith* layer, and b_i is the offset vector.

Convolution neural network uses the full connection layer to classify the extracted features and obtain the probability distribution λ of the raw sample by the alternating operation of multiple convolutions and subsampling layers. Its mathematical model can be expressed as:

$$\lambda(i) = Map(C = c_i | P_0;\ (W, b)) \tag{4}$$

where λ denotes the feature expression, c_i represents the *ith* label class, and P_0 represents the raw input sample.

The training objective of CNN is to minimize the loss function $L(W, b)$ of the network. Meanwhile, in order to alleviate the over-fitting problem, the final loss function $F(W, b)$ is usually constrained by the norm and the over-fitting intensity is adjusted by the parameter ε:

$$F(W, b) = L(W, b) + \frac{\varepsilon}{2} W^T W \tag{5}$$

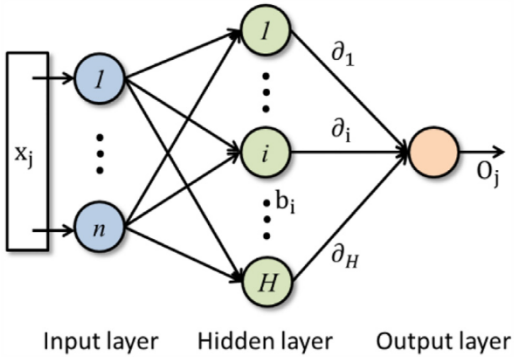

Fig. 2. Structure diagram of extreme learning machine.

During the training process, CNN usually uses the gradient descent method to optimize the parameter (W, b), and through the learning rate α to control the intensity of back-propagation.

$$W_i = W_i - \alpha \frac{\partial E(W, b)}{\partial W_i} \text{ and } b_i = b_i - \alpha \frac{\partial E(W, b)}{\partial b_i} \tag{6}$$

2.4. Extreme learning machine

Extreme learning machine (ELM) is an easy-to-use and effective single-hidden layer feed forward neural network learning algorithm proposed by Huang et al. (2006). ELM does not need to adjust the input weights of network and hidden unit bias in the implementation process, only need to set the number of hidden nodes of the network, and can produce a unique optimal solution, thus has the advantages of fast learning speed and good generalization performance. Since ELM has these characteristics, we choose it as a basic classifier in our RPIFSE model.

An ELM structure with H hidden nodes and N input samples (X_i, Y_i) is shown in Fig. 2, and the formula is expressed as:

$$\sum_{i=1}^{H} \partial_i f\left(W_i \cdot X_j + b_i\right) = O_j,\ \ j = 1, \ldots, N \tag{7}$$

where $X_i = [x_{i1}, x_{i2,\ldots,} x_{in,}]^T \in R^n$ is the input data, $Y_i = [y_{i1}, y_{i2}, \ldots, y_{im}]^T \in R^m$ is the label, $W_i = [w_{i1}, w_{i2}, \ldots, w_{in}]^T$ represents the input weight, ∂_i represents the output weight, b_i means the offset of the *ith* hidden layer, $f(x)$ means the activation function, and $W_i \cdot X_j$ represents the inner product of W_i and X_j.

The learning goal of extreme learning machine is to minimize the output error, that is,$\sum_{j=1}^{n}\|O_j - Y_j\| = 0$. Therefore, the ELM needs to find the appropriate parameter W_i, $\cdot\,\partial_i$ and b_i to make $\sum_{i=1}^{H}\partial_i f(W_i \cdot X_j + b_i) = Y_j$ The formula can also be expressed as $F\partial = Y$. In order to achieve these objectives, we hope to get $\widehat{W}_i$, $\widehat{b}_i$ and $\widehat{\partial}_i$ when training ELM, so that

$$\left\|F\left(\widehat{W}_i, \widehat{b}_i\right)\widehat{\partial}_i - Y\right\| = \min_{W,b,\partial}\|F(W_i, b_i)\partial_i - Y \;\|\; i = 1, 2, \cdots, H \tag{8}$$

This is equivalent to minimizing the loss function

$$E = \sum_{j=1}^{N}\left(\sum_{i=1}^{H}\partial_i f\left(W_i \cdot X_j + b_i\right) - Y_j\right)^2 \tag{9}$$

In the ELM algorithm, once the input weight W_i and the offset b_i of the hidden layer are determined, the output matrix F is uniquely determined. Thus, the task of training the hidden neural network is transformed into solving the linear equation $F\partial = Y$, and it can be proved that the solution of the equation is minimal and unique.

3. Results and discussion

3.1. Ensemble strategy of RPIFSE

In this study, we use ensemble strategy to construct RPIFSE model. The core of the ensemble strategy is to find a learning algorithm which is little better than random guessing, by collecting the multiple computation results of the algorithm to achieve the purpose of improving the accuracy rate. Here, considering the importance of feature, we use the method of interference feature space to achieve the purpose of diversity of classification results.

High-dimensional features can contain more information and more comprehensively reflect reality, but inevitably carry redundant and noisy information. So we introduce χ^2 statistical method to calculate the weight of the feature, and select the features with high weights according to different ratios to construct data sets. This not only eliminates noise information, but also increases the diversity of data sets. The weight of the feature F can be calculated as follows:

$$\chi^2 = \sum_{i=1}^{n}\sum_{j=1}^{2}\frac{\left(\alpha_{ij} - \beta_{ij}\right)^2}{\beta_{i,j}} \tag{10}$$

where n is the number of values in feature F, α_{ij} is the count of the value θ_i in feature F and belong to class p_j, defined as:

$$\alpha_{ij} = count\left(F = \theta_i \;and\; P = p_j\right) \tag{11}$$

$\beta_{i,j}$ is the expected value of θ_i and p_j, defined as:

$$\beta_{i,j} = \frac{count\,(F = \theta_i) \;\times count\left(P = p_j\right)}{N} \tag{12}$$

where $count\,(F = \theta_i)$ is the number of samples in the feature F value is θ_i, $count\,(P = p_j)$ is the number of samples in the class P value is p_j, and N is the total number of samples in the data set. In the experiment, we selected data sets of 100%, 90%, and 80% to construct data sets.

Convolution neural network is the feedforward neural network, which can automatically extract the advanced features representing the raw sample by layer iteration. If it is given to the label with the data, it can also be fine-tuned to more accurately extract the features of the sample. Therefore, according to the characteristics of convolution neural network, we preserved the features of fine-tuning after and before. In addition, we also joined the raw sample. In this way, combined with the three selection ratios, we constructed a total of nine different data sets to predict the interaction among RNAs and proteins, namely, AFT-10F, BFT-10F, RAW-10F, AFT-9F, BFT-9F, RAW-9F, AFT-8F, BFT-8F, RAW-8F.

In the integration of the results of each classifier, we use a weighted voting method with higher recognition rate. The method first sets a different weight for each base classifier, and then selects the highest scoring class as the class of samples after synthesizing the results and weights of all classifiers. Suppose that the prediction accuracy of the *ith* classifier on the training set is P_i, and the weight of the classifier can be calculated as follows:

$$W_i = \frac{\log P_i/(1 - P_i)}{\sum_{i=1}^{n}\log P_i/(1 - P_i)} \tag{13}$$

Study (Kuncheva, 2005) demonstrates that the weight calculation method designed by formula 13 can make full use of the prior information of each classifier and maximize the recognition accuracy of the ensemble system. The flow chart for RPIFSE is shown in Fig. 3.

We summarize the results of the basic classifiers and the final ensemble results on RPI1807 data set in Table 2. From the table we can see that the result of the ensemble is higher than the result of any one of the basic classifiers. This shows that our ensemble strategy is feasible and competitive. From the performance of each basic classifier, the result of using all the features on RPI1807 data set is higher than those using 90%, while the result of using 90% features is higher than those using 80%. This indicates that the RPI1807 data set contains less noise information. However, we found on RPI369 data set that the result of using 80% features is the highest, indicating that the RPI369 data set contains some noise information. In addition, we also find that the result of the after fine-tuning of the convolution neural network is higher than those before the fine-tuning, and the original feature produces the worst result. This suggests that the fine-tuning of convolution neural network does help to improve the expression of features.

3.2. Evaluation criteria

In this experiment, we used 5-fold cross-validation to evaluate the performance of RPIFSE. The method first divides the training set T into 5 subsets $T_1, \ldots, T_5$ which are approximately equal in size and disjoint in each other. For any subset T_i, use the $T - T_i$ training classifier and then test the generated classifier with T_i. This is repeated until all subsets have been used for testing. Finally, the average of the 5 tests is calculated as the result of the classifier. Here, we follow the widely used criteria to evaluate RPIFSE, including accuracy (Accu.), sensitivity (Sen.), specificity (Spec.), precision (Prec.) and Matthews correlation coefficient (MCC). They are calculated as:

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

$$Sen. = \frac{TP}{TP + FN} \tag{15}$$

$$Spec. = \frac{TN}{TN + FP} \tag{16}$$

$$Prec. = \frac{TP}{TP + FP} \tag{17}$$

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

where TP, FP,TN,FN indicate the number of true positive, false positive, true negative and false negative. In addition, the receiver operating characteristic (ROC) (Swets, 1988; Zweig and Campbell, 1993)

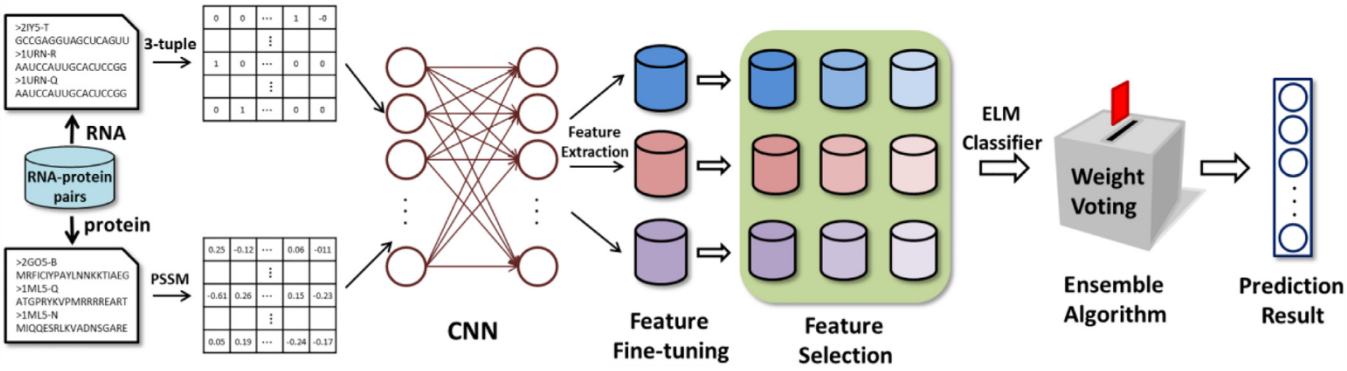

Fig. 3. The flowchart of the proposed RPIFSE.

Table 2
The results of the basic classifiers and the final ensemble results on RPI1807 data set.

Basic Classifier	Accu.	Sen.	Spec.	Prec.	MCC	WEIGHT
AFT-10F	9.78e−01	9.82e−01	9.73e−01	9.79e−01	9.55e−01	1.61e−01
BFT-10F	9.76e−01	9.79e−01	9.72e−01	9.77e−01	9.51e−01	1.58e−01
RAW-10F	6.61e−01	8.02e−01	4.98e−01	6.76e−01	3.27e−01	2.84e−02
AFT-9F	9.75e−01	9.77e−01	9.73e−01	9.78e−01	9.49e−01	1.56e−01
BFT-9F	9.73e−01	9.76e−01	9.69e−01	9.75e−01	9.46e−01	1.53e−01
RAW-9F	6.52e−01	8.60e−01	4.02e−01	6.47e−01	3.12e−01	2.67e−02
AFT-8F	9.73e −01	9.69e−01	9.78e−01	9.81e−01	9.45e−01	1.52e−01
BFT-8F	9.66e−01	9.71e−01	9.60e−01	9.66e−01	9.31e−01	1.42e−01
RAW-8F	6.28e−01	8.39e−01	3.69e−01	6.31e−01	2.29e−01	2.23e−02
Ensemble Result	9.90e−01	9.93e−01	9.86e−01	9.88e−01	9.80e−01	1.00e−00

Table 3
5-fold cross-validation results performed by RPIFSE on RPI369 data set.

Test set	Accu.	Sen.	Spec.	Prec.	MCC	AUC
1	9.59e−01	9.76e−01	9.38e−01	9.53e−01	9.17e−01	9.62e−01
2	9.05e−01	9.32e−01	8.77e−01	8.85e−01	8.11e−01	8.93e−01
3	8.84e−01	8.77e−01	8.92e−01	8.89e−01	7.69e−01	8.72e−01
4	9.39e−01	8.92e−01	9.76e−01	9.67e−01	8.77e−01	9.12e−01
5	9.07e−01	9.05e−01	9.08e−01	9.05e−01	8.13e−01	8.95e−01
Average	**9.19e−01**	**9.17e−01**	**9.18e−01**	**9.20e−01**	**8.37e−01**	**9.07e−01**
	±2.98e−02	**±3.90e−02**	**±3.93e−02**	**±3.77e−02**	**±5.90e−02**	**±3.39e−02**

curve and the area under the ROC curve (AUC) (Bradley, 1997) are also calculated as evaluation criteria. The ROC curve establishes a series of different thresholds based on the binary classification problem. It is shown by plotting the ratio of true positive rate (sensitivity) to false positive rate (1-specificity). The range of AUC is from 0.5 to 1 and the greater the value, the better the performance of classifier.

3.3. Assessment of prediction ability

We evaluated the predictability of RPIFSE by collecting four data sets including RPI369, RPI2241, RPI488 and RPI1807. We can see from Table 3 that the accuracy, sensitivity, specificity, precision, MCC and AUC of RPIFSE on RPI369 data set are 91.87%, 91.66%, 91.79%, 91.97%, 83.73% and 90.66%, respectively. Their standard deviations are 2.98%, 3.90%, 3.93%, 3.77%, 5.90% and 3.39%, respectively. Table 4 lists the RPIFSE cross-validation results on RPI2241 data set, from which we can see that the average accuracy is 89.74%, the sensitivity is 91.80%, the specificity is 87.72%, the precision is 88.17%, the MCC is 79.56%, and the AUC is 89.34%. Their standard deviations are 0.82%, 0.72%, 1.69%, 1.96%, 1.55%, and 0.13%, respectively. Table 5 summarizes the results of the RPIFSE on RPI488 data set, with accuracy, sensitivity, specificity, precision, MCC and AUC of 97.76%, 95.90%, 99.57%, 99.60%, 95.60%, and 97.79%, respectively. Their standard deviations are 1.93%, 3.96%, 0.97%, 0.89%, 3.72% and 1.58%. The results of RPI1807 data set are shown in Table 6, with the accuracy of 98.98%, the sensitivity of 99.34%, the specificity of 98.58%, the precision of 98.82%, the MCC of 97.95%, and the AUC of 98.88%. Their standard deviations are 0.42%, 0.51%, 1.22%, 1.18%, 0.80% and 0.57%, respectively. Figs. 4–7

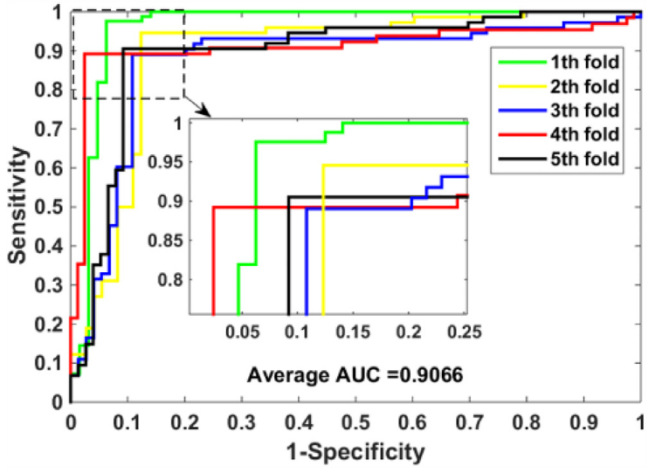

Fig. 4. ROC curves performed by RPIFSE on RPI369 data set.

Table 4
5-fold cross-validation results performed by RPIFSE on RPI2241 data set.

Test set	Accu.	Sen.	Spec.	Prec.	MCC	AUC
1	8.93e−01	9.13e−01	8.73e−01	8.78e−01	7.86e−01	8.83e−01
2	9.02e−01	9.14e−01	8.89e−01	8.99e−01	8.03e−01	9.05e−01
3	9.04e−01	9.13e−01	8.95e−01	9.01e−01	8.08e−01	9.02e−01
4	8.85e−01	9.21e−01	8.51e−01	8.52e−01	7.73e−01	8.76e−01
5	9.03e−01	9.29e−01	8.78e−01	8.79e−01	8.08e−01	9.01e−01
Average	**8.97e−01**	**9.18e−01**	**8.77e−01**	**8.82e−01**	**7.96e−01**	**8.93e−01**
	±8.20e−03	**±7.20e−03**	**±1.69e−02**	**±1.96e−02**	**±1.55e−02**	**±1.30e−03**

Table 5
5-fold cross-validation results performed by RPIFSE on RPI488 data set.

Test set	Accu.	Sen.	Spec.	Prec.	MCC	AUC
1	9.69e−01	9.61e−01	9.78e−01	9.80e−01	9.38e−01	9.73e−01
2	9.79e−01	9.64e−01	1.00e−00	1.00e−00	9.59e−01	9.82e−01
3	1.00e−00	1.00e−00	1.00e−00	1.00e−00	1.00e−00	1.00e−00
4	9.90e−01	9.76e−01	1.00e−00	1.00e−00	9.79e−01	9.78e−01
5	9.50e−01	8.94e−01	1.00e−00	1.00e−00	9.04e−01	9.56e−01
Average	**9.78e−01**	**9.59e−01**	**9.96e−01**	**9.96e−01**	**9.56e−01**	**9.78e−01**
	±1.93e−02	**±3.96e−02**	**±9.70e−03**	**±8.90e−03**	**±3.72e−02**	**±1.58e−02**

Table 6
5-fold cross-validation results performed by RPIFSE on RPI1807 data set.

Test set	Accu.	Sen.	Spec.	Prec.	MCC	AUC
1	9.83e−01	1.00e−00	9.65e−01	9.68e−01	9.67e−01	9.81e−01
2	9.94e−01	9.92e−01	9.97e−01	9.97e−01	9.88e−01	9.96e−01
3	9.91e−01	9.89e−01	9.93e−01	9.94e−01	9.81e−01	9.93e−01
4	9.89e−01	9.89e−01	9.90e−01	9.92e−01	9.78e−01	9.87e−01
5	9.92e−01	9.97e−01	9.85e−01	9.90e−01	9.84e−01	9.87e−01
Average	**9.90e−01**	**9.93e−01**	**9.86e−01**	**9.88e−01**	**9.80e−01**	**9.89e−01**
	±4.20e−03	**±5.10e−03**	**±1.22e−02**	**±1.18e−02**	**±8.00e−03**	**±5.70e−03**

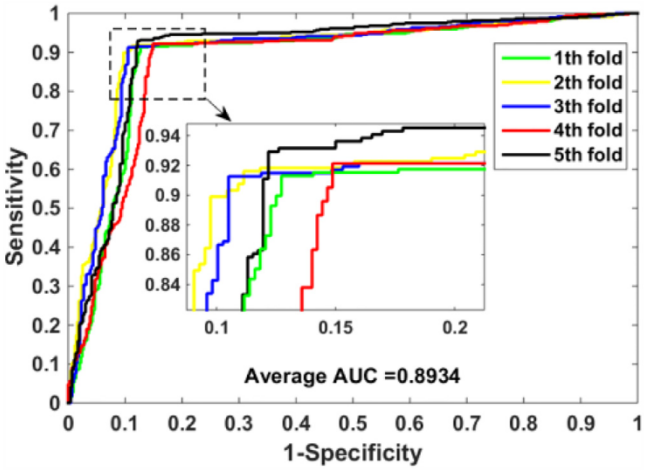

Fig. 5. ROC curves performed by RPIFSE on RPI2241 data set.

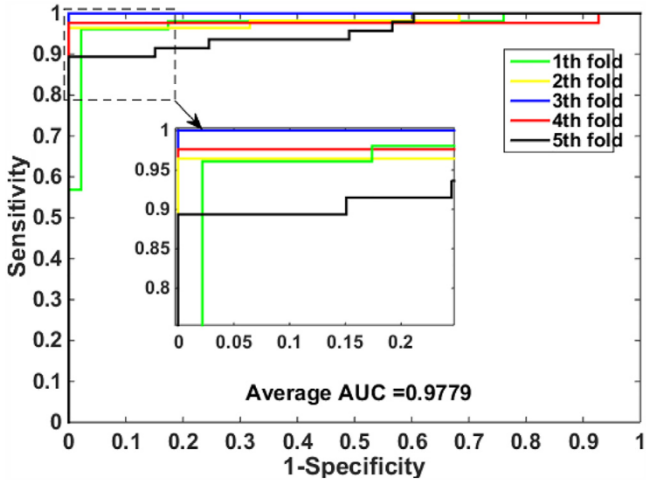

Fig. 6. ROC curves performed by RPIFSE on RPI488 data set.

show the ROC curves generated on RPI369, RPI2241, RPI488 and RPI1807 data sets, respectively.

3.4. Comparison between RPIFSE and SVM-based method

Support vector machine is a widely used two-classification model, which is commonly used for pattern recognition, classification, and regression analysis (Cao et al., 2010, 2011; Smola and Lkopf, 2004). To validate the performance of RPIFSE, we compare it with SVM-based method. Since SVM is not an ensemble classifier, we choose the best feature set AFT-10F to predict the RNA-protein interaction in the experiment. Table 7 shows the results of 5-fold cross-validation of RPIFSE and SVM-based method on data sets RPI369, RPI2241, RPI488 and RPI1807. On RPI369 data set, the SVM-based method achieved average accuracy, sensitivity, specificity, precision, MCC and AUC are 81.84%, 75.03%, 88.37%, 86.44%, 64.05% and 81.32%, the standard deviation are 3.22%, 6.02%, 2.88%, 3.64%, 6.19% and 3.59%, respectively. On RPI2241 data set, the average accuracy of 5-fold cross-validation of SVM-based method is 82.53%, the sensitivity is 77.21%, the specificity is 87.95%, the precision is 86.50%, the MCC is 65.53%, and the AUC is 81.97%. Their standard deviations are 4.01%, 2.76%, 5.95%, 6.67%, 8.42% and 4.85%, respectively. On RPI488 data set, the average accuracy, sensitivity, specificity, precision, MCC and AUC of SVM-based method are 93.43%, 90.74%, 96.43%, 95.95%, 87.03% and 93.72%, the standard deviation

Table 7
Comparison between RPIFSE and SVM-based method on RPI369, RPI2241, RPI488 and RPI1807 data sets.

Data set	Method	Accu.	Sen.	Spec.	Prec.	MCC	AUC
RPI369	RPIFSE	**9.19e−01**	**9.17e−01**	**9.18e−01**	**9.20e−01**	**8.37e−01**	**9.07e−01**
	SVM-based	8.18e−01	7.50e−01	8.84e−01	8.64e−01	6.41e−01	8.13e−01
RPI2241	RPIFSE	**8.97e−01**	**9.18e−01**	**8.77e−01**	**8.82e−01**	**7.96e−01**	**8.93e−01**
	SVM-based	8.25e−01	7.72e−01	**8.80** e−01	8.65e−01	6.55e−01	8.20e−01
RPI488	RPIFSE	**9.78e−01**	**9.59e−01**	**9.96** e−**01**	**9.96e−01**	**9.56e−01**	**9.78e−01**
	SVM-based	9.34e−01	9.07e−01	9.64e−01	9.60e−01	8.70e−01	9.37e−01
RPI1807	RPIFSE	**9.90e−01**	**9.93e−01**	**9.86e−01**	**9.88e−01**	**9.80e−01**	**9.89e−01**
	SVM-based	9.59e−01	9.59e−01	9.60e−01	9.67e−01	9.18e−01	9.59e−01

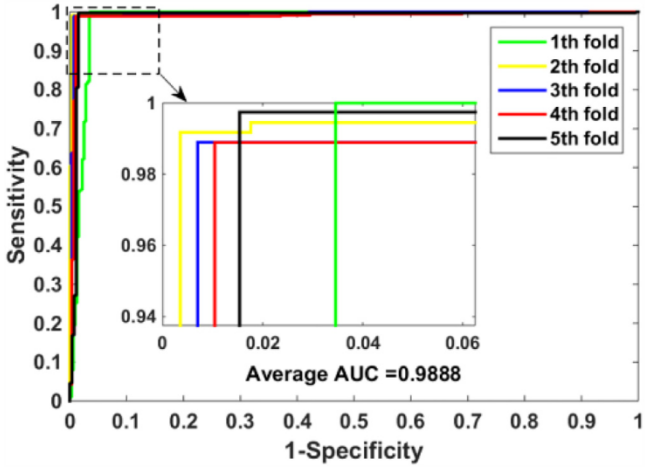

Fig. 7. ROC curves performed by RPIFSE on RPI1807 data set.

Table 8
Performance comparison of different methods on RPI369, RPI2241, RPI488 and RPI1807 data sets.

Data set	Method	Accu.	Prec.	AUC
RPI369	RPI-Pred	**9.20e−01**	8.90e−01	**9.50e−01**
	RPISeq-RF	7.62e−01	7.30e−01	8.50e−01
	RPISeq-SVM	7.28e−01	7.30e−01	8.10e−01
	IPMiner	7.52e−01	7.13e−01	7.73e−01
	RPIFSE	9.19e−01	**9.20e−01**	9.07e−01
RPI2241	RPI-Pred	8.40e−01	8.80e−01	8.90e−01
	RPISeq-RF	8.96e−01	**8.90e−01**	**9.70e−01**
	RPISeq-SVM	8.71e−01	8.70e−01	9.20e−01
	IPMiner	8.24e−01	8.36e−01	9.06e−01
	RPIFSE	**8.97e−01**	8.82e−01	8.93e−01
RPI488	IPMiner	8.91e−01	9.45e−01	9.14e−01
	RPIFSE	**9.78e−01**	**9.96e−01**	**9.78e−01**
RPI1807	RPI-Pred	9.30e−01	9.40e−01	9.70e−01
	IPMiner	9.86e−01	9.78e−01	**9.98e−01**
	RPIFSE	**9.90e−01**	**9.88e−01**	9.89e−01

are 2.81%, 4.01%, 2.65%, 2.98%, 5.46% and 2.73%, respectively. On RPI1807 data set, the average accuracy of 5-fold cross-validation of SVM-based method is 95.93%, the sensitivity is 95.94%, the specificity is 95.95%, the precision is 96.67%, the MCC is 91.76%, and the AUC is 95.93%. Their standard deviations are 1.24%, 0.56%, 2.14%, 1.99%, 2.48%, and 1.36%, respectively. We can see from the comparison results that the performance of RPIFSE is generally better than the SVM-based approach. Especially, the accuracy of RPIFSE on RPI369, RPI2241, RPI488 and RPI1807 data sets is 10.03%, 7.21%, 4.33% and 3.05% higher than the SVM-based approach, respectively.

3.5. Comparison with other methods

In order to verify the performance of RPIFSE, we try to compare it with other methods with the same evaluation criteria on the same data set. For comparison of the same data set, we chose the RPI-Pred (Suresh et al., 2015), RPISeq-RF (Muppirala et al., 2011), RPISeq-SVM (Muppirala et al., 2011) and IPMiner (Pan et al., 2016) methods. It should be noted that in these methods, only IPMiner did experiments on RPI488 data set, and RPISeq-RF and RPISeq-SVM did not do experiments on RPI1807 data set, so we could not list them in the summary table. Since the criteria used by the researchers are not exactly the same, we here only list the common evaluation criteria in their research, including accuracy (Accu.), precision (Prec.) and AUC.

Table 8 summarizes the performance of different methods on RPI369, RPI2241, RPI488 and RPI1807 data sets. In terms of accuracy, RPIFSE achieves the best results on three of those four data sets, 0.13% lower than the RPI-Pred method on RPI369 data set, and 0.14%, 8.66% and 0.38% higher than the second score on RPI2241, RPI488 and RPI1807 data sets, respectively. In terms of precision, RPIFSE also achieves three best results, 0.83% lower than the RPISeq-RF method on RPI2241 data set, and 2.97%, 5.10% and 1.02% higher than the second score on RPI369, RPI488 and RPI1807 data sets, respectively. In terms of AUC, RPIFSE scores generally, 6.40% higher than the second score on RPI488 data set, 4.34%, 7.66% and 0.92% lower than the highest score on RPI369, RPI2241 and RPI1807 data sets, respectively.

3.6. Verification on an independent data set

In order to verify the versatility of RPIFSE, we try to use RPI369 data set as the training sample to predict the interaction among RNA and protein on the independent data set NPInter2.0. NPInter2.0 data set contains 10,412 pairs of RNA-protein interactions that can be divided into six organisms. Specifically, these organisms are *Caenorhabditis elegans, Drosophila melanogaster, Escherichia coli, Homo sapiens, Mus musculus* and *Saccharomyces cerevisiae*, and the number of pairs of RPI they contain is 36, 91, 202, 6975, 2198 and 910, respectively. In the experiment, we use the same ensemble strategy to predict the RNA-protein interaction among the six organisms. As shown in Table 9, RPIFSE correctly predicted the number of pairs of RPI is 33, 88, 188, 6551, 2189 and 894, with accuracy of 91.67%, 96.70%, 93.07%, 93.92%, 99.59% and 98.24%, respectively. Finally, RPIFSE correctly predicted 9943 protein pairs with an accuracy of 95.50% on NPInter2.0 data set.

3.7. The application of RPIFSE in establishing ncRNA-protein network

Visualization model can give people more intuitive feelings, so we use the prediction results of RPIFSE to establish the ncRNA-protein network of the NPInter2.0 data set. In the construction process, we use the Markov clustering algorithm (Van Dongen) for network clustering, and the nodes in the network represent the ncRNAs or proteins and the edges represent the interaction among them. Taking the *Drosophila melanogaster* of the NPInter2.0 data set as an example, the RPIFSE correctly predicted 88 protein pairs. Based on this prediction result, we constructed the ncRNA-protein

Table 9
The predicted results of trained model from RPI369 on NPInter2.0 data set.

Organism	The number of pairs of RNA-protein interaction	Correctly predict the number of pairs of RNA-protein interaction	Accuracy
Caenorhabditis elegans	36	33	9.17e−01
Drosophila melanogaster	91	88	9.67e−01
Escherichia coli	202	188	9.31e−01
Homo sapiens	6975	6551	9.39e−01
Mus musculus	2198	2189	9.96e−01
Saccharomyces cerevisiae	910	894	9.82e−01
Total	10,412	9943	9.55e−01

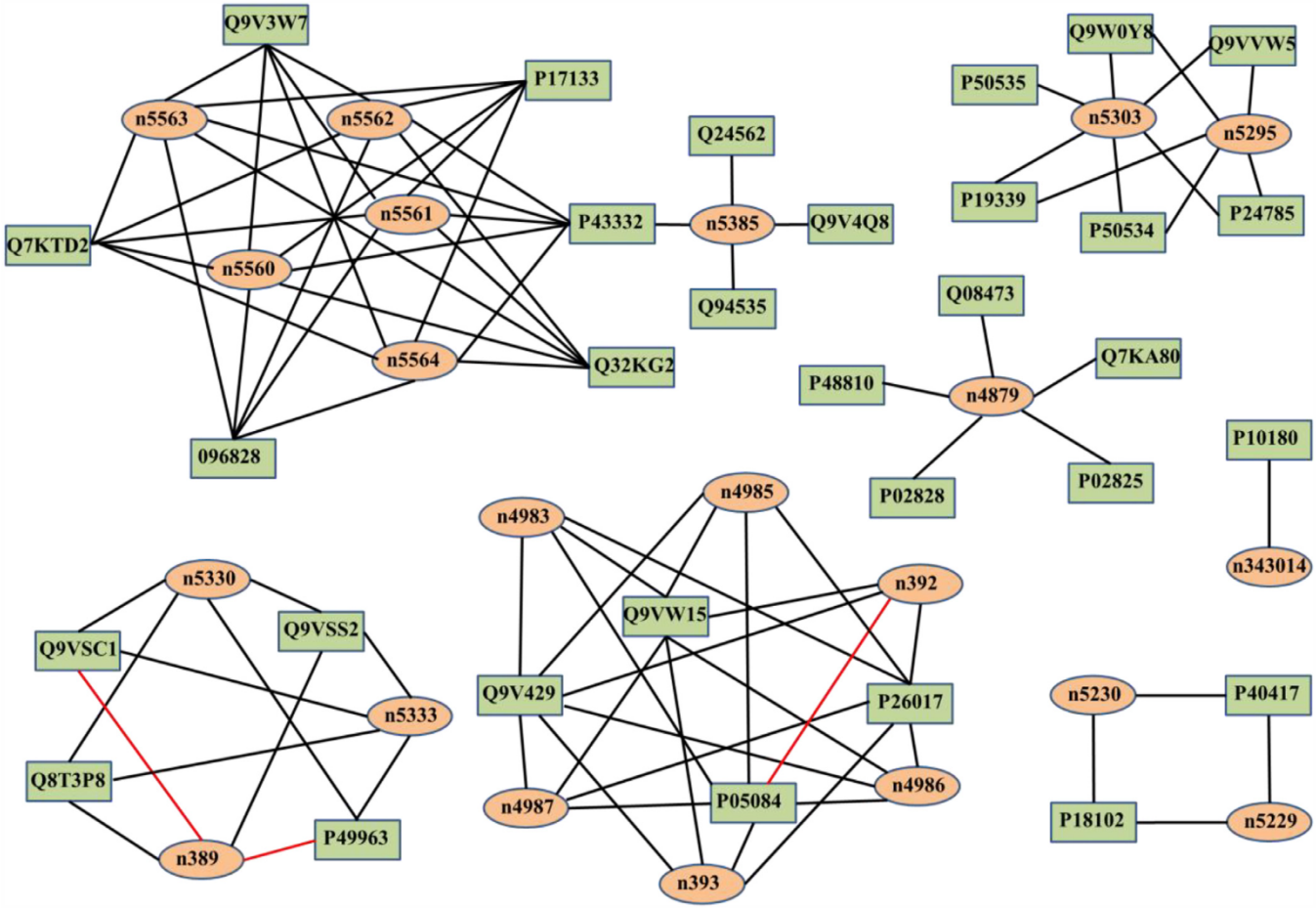

Fig. 8. The *Drosophila melanogaster* networks constructed based on interaction pairs predicted by the RPIFSE. The oval orange and rectangular green the nodes represent the RNA and protein, respectively. The black and red edges indicate the correct and the erroneous prediction of the ncRNA-protein interactions, respectively. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

network model, as shown in Fig. 8. Building a visualized network model helps to represent the relationship among RNAs and proteins, and helps to derive potential RNA and protein interaction.

4. Conclusion

In this paper, we propose a new ensemble method RPIFSE based on RNA and protein sequence information to predict RNA-protein interaction. The method generates multiple data sets by selecting whether the convolution neural network is fine-tuned and the features of different weights. These data sets are sent to the ELM base classifier for classification, and then select the most likely categories as the final prediction result by weighted voting method. In the experiments, we evaluated the performance of RPIFSE on RPI369, RPI2241, RPI488 and RPI1807 data sets, respectively. In order to have a clearer understanding of RPIFSE, we compared it to state-of-the-art SVM classifiers and other methods. In addition, we also predicted the RNA-protein interaction of the independent data set NPInter2.0, and plotted the ncRNA-protein network according to the predicted results. These excellent experimental results show that RPIFSE is robust and competitive in predicting the interaction of RNA and protein. In the future work, we will continue to improve the ensemble method in anticipation of better results in RNA-protein interaction prediction.

Competing interests

The authors declare that they have no competing interests.

Acknowledgment

This work is supported by the National Science Foundation of China, under Grants 61702444. The authors would like to thank all anonymous reviewers for their constructive advices.

Conflicts of interest

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

References

Altschul, S.F., Madden, T.L., Schaffer, A.A., Zhang, J., Zhang, Z., Miller, W., Lipman, D.J., 1997. Gapped BLAST and PSI-BLAST: a new generation of protein database search programs. Nucleic Acids. Res. 25, 3389–3402. doi:10.1093/nar/25.17.3389.

Anbarasu, A., Sethumadhavan, R., 2007. Exploring the role of cation-pi interactions in glycoproteins lipid-binding proteins and RNA-binding proteins. J. Theor. Biol. 247, 346–353.

Bellucci, M., Agostini, F., Masin, M., Tartaglia, G.G., 2011. Predicting protein associations with long noncoding RNAs. Nat. Methods 8, 444–445.

Bradley, A.P., 1997. The use of the area under the ROC curve in the evaluation of machine learning algorithms. Pattern Recognit. 30, 1145–1159.

Cao, D.S., Xu, Q.S., Liang, Y.Z., Xian, C., Li, H.D., 2010. Prediction of aqueous solubility of druglike organic compounds using partial least squares, back-propagation network and support vector machine. J. Chemom. 24, 584–595.

Cao, D.S., Liang, Y.Z., Xu, Q.S., Hu, Q.N., Zhang, L.X., Fu, G.H., 2011. Exploring nonlinear relationships in chemical data using kernel-based methods. Chemom. Int. Lab. Syst. 107, 106–115.

Chen, X.-W., Jeong, J.C., 2009. Sequence-based prediction of protein interaction sites with an integrative method. Bioinformatics 25, 585–591. doi:10.1093/bioinformatics/btp039.

Chen, Y., Varani, G., 2005. Protein families and RNA recognition. FEBS J. 272, 2088.

Cheng-Wei, C., Emily Chia-Yu, S., Jenn-Kang, H., Ting-Yi, S., Wen-Lian, H., 2008. Predicting RNA-binding sites of proteins using support vector machines and evolutionary information. Bmc Bioinform. 12, S6.

Coelho, E.D., Cruz, I.N., Santiago, A., Oliveira, J.L., Dourado, A., Arrais, J.P., 2017. A Sequence-Based Mesh Classifier for the Prediction of Protein-Protein Interactions.

Cooper, T.A., Wan, L., Dreyfuss, G., 2011. RNA and disease. Cell 136, 777–793.

Fujishima, K., Komasa, M., Kitamura, S., Suzuki, H., Tomita, M., Kanai, A., 2007. Proteome-Wide Prediction of Novel DNA/RNA-Binding Proteins Using Amino Acid Composition and Periodicity in the Hyperthermophilic Archaeon Pyrococcus furiosus. DNA Research,14,3(2007-6-15) 14, 91–102.

Gao, Z.G., Wang, L., Xia, S.X., You, Z.H., Yan, X., Zhou, Y., 2016. Ens-PPI: a novel ensemble classifier for predicting the interactions of proteins using Autocovariance transformation from PSSM. Biomed. Res. Int. 8. doi:10.1155/2016/4563524.

Gribskov, M., McLachlan, A.D., Eisenberg, D., 1987. Profile analysis: detection of distantly related proteins. In: Proceedings of the National Academy of Sciences of the United States of America, 84, pp. 4355–4358. doi:10.1073/pnas.84.13.4355.

Guo, Y., Yu, L., Wen, Z., Li, M., 2008. Using support vector machine combined with auto covariance to predict protein–protein interactions from protein sequences. Nucleic. Acids. Res. 36, 3025–3030.

Huang, G.B., Zhu, Q.Y., Siew, C.K., 2006. Extreme learning machine: theory and applications. Neurocomputing 70, 489–501.

Jones, D.T., 1999. Protein secondary structure prediction based on position-specific scoring matrices. J. Mol. Biol. 292, 195–202. doi:10.1006/jmbi.1999.3091.

Jones, D.T., Ward, J.J., 2003. Prediction of disordered regions in proteins from position specific score matrices. Proteins-Struct. Funct. Bioinform. 53, 573–578. doi:10.1002/prot.10528.

Kalchbrenner, N., Grefenstette, E., Blunsom, P., 2014. A convolutional neural network for modelling sentences. Eprint Arxiv 1.

Ke, A., Doudna, J.A., 2004. Crystallization of RNA and RNA–protein complexes. Methods 34, 408.

Kumar, M., Gromiha, M.M., Raghava, G.P., 2011. SVM based prediction of RNA-binding proteins using binding residues and evolutionary information. J. Mol. Recognit. 24, 303.

Kuncheva, L.I., 2005. Combining pattern classifiers: methods and algorithms. Technometrics 47, 517–518.

Lewis, B.A., Walia, R.R., Terribilini, M., Ferguson, J., Zheng, C., Honavar, V., Dobbs, D., 2011. PRIDB: a protein-RNA interface database. Nucleic. Acids. Res. 39, 277–282.

Lu, Q., Ren, S., Lu, M., Zhang, Y., Zhu, D., Zhang, X., Li, T., 2013. Computational prediction of associations between long non-coding RNAs and proteins. BMC Genom. 14, 651.

Lukong, K.E., Chang, K.W., Khandjian, E.W., Richard, S., 2008. RNA-binding proteins in human genetic disease. Trends Genet. 24, 416.

Muppirala, U.K., Honavar, V.G., Dobbs, D., 2011. Predicting RNA-protein interactions using only sequence information. BMC Bioinform. 12, 489.

Nashimoto, M., 2001. The RNA/protein symmetry hypothesis: experimental support for reverse translation of primitive proteins. J. Theor. Biol. 209, 181.

Pai, P.P., Dash, T., Mondal, S., 2017. Sequence-based discrimination of protein-RNA interacting residues using a probabilistic approach. J. Theor. Biol. 418, 77.

Pan, X., Fan, Y.X., Yan, J., Shen, H.B., 2016. IPMiner: hidden ncRNA-protein interaction sequential pattern mining with stacked autoencoder for accurate computational prediction. BMC Genom. 17, 582.

Rajagopal, S., Vishveshwara, S., 2005. Short hydrogen bonds in proteins. FEBS J. 272, 1819–1832.

Rice, P., Longden, I., Bleasby, A., 2000. EMBOSS: The European molecular biology open software suite. Trends Genet. 16, 276–277. doi:10.1016/s0168-9525(00)02024-2.

Scott, L.G., Hennig, M., 2008. RNA structure determination by NMR. Methods Mol. Biol. 452, 29.

Smola, A.J., Lkopf, B., 2004. A tutorial on support vector regression. Stat. Comput. 14, 199–222.

Suresh, V., Liu, L., Adjeroh, D., Zhou, X., 2015. RPI-Pred: predicting ncRNA-protein interaction using sequence and structural information. Nucleic. Acids. Res. 43, 1370–1379.

Swets, J.A., 1988. Measuring the accuracy of diagnostic systems. Science 240, 1285.

Wang, L., You, Z.H., Chen, X., Li, J.Q., Yan, X., Zhang, W., Huang, Y.A., 2016. An ensemble approach for large-scale identification of protein- protein interactions using the alignments of multiple sequences. Oncotarget 8, 5149.

Wang, L., You, Z.-H., Xia, S.-X., Chen, X., Yan, X., Zhou, Y., Liu, F., 2017a. An improved efficient rotation forest algorithm to predict the interactions among proteins. Soft Computing 1–9.

Wang, L., You, Z.-H., Xia, S.-X., Liu, F., Chen, X., Yan, X., Zhou, Y., 2017b. Advancing the prediction accuracy of protein-protein interactions by utilizing evolutionary information from position-specific scoring matrix and ensemble classifier. J. Theor. Biol. 418, 105–110. doi:10.1016/j.jtbi.2017.01.003.

Xu, R., Zhou, J., Wang, H., He, Y., Wang, X., Liu, B., 2015. Identifying DNA-binding proteins by combining support vector machine and PSSM distance transformation. BMC Syst. Biol. 9, S10.

Yu, H.J., Huang, D.S., 2012. Novel graphical representation of genome sequence and its applications in similarity analysis. Phys. Stat. Mech. Appl. 391, 6128–6136.

Yu, X., Cao, J., Cai, Y., Shi, T., Li, Y., 2006. Predicting rRNA-, RNA-, and DNA-binding proteins from primary structure with support vector machines. J. Theor. Biol. 240, 175–184.

Yuan, J., Wu, W., Xie, C., Zhao, G., Zhao, Y., Chen, R., 2014. NPInter v2.0: an updated database of ncRNA interactions. Nucleic. Acids. Res. 42, 104–108.

Zahiri, J., Yaghoubi, O., Mohammad-Noori, M., Ebrahimpour, R., Masoudi-Nejad, A., 2013. PPIevo: Protein–protein interaction prediction from PSSM based evolutionary information. Genomics 102, 237–242.

Zhang, L., Zhang, C., Gao, R., Yang, R., Song, Q., 2016. Prediction of aptamer-protein interacting pairs using an ensemble classifier in combination with various protein sequence attributes. BMC Bioinform. 17, 225.

Zweig, M.H., Campbell, G., 1993. Receiver-operating characteristic (ROC) plots: a fundamental evaluation tool in clinical medicine. Clin. Chem. 39, 561–577.