applied sciences

MDPI

Article

A Deep Model for Species-Specific Prediction of Ribonucleic-Acid-Binding Protein with Short Motifs

Zhi-Sen Wei [1,2,*], Jun Rao [1] and Yao-Jin Lin [1]

[1] School of Computer Science, Minnan Normal University, Zhangzhou 363000, China
[2] Fujian Province Universities Key Laboratory of Data Science and Intelligent Application, Minnan Normal University, Zhangzhou 363000, China
* Correspondence: zs.wei@foxmail.com

Highlights:

What are the main findings?

- A novel species-specific RBP predictor based on a convolutional neural network with short motifs,
- State-of-the-art performance based on simple sequence features,
- Difference in discriminative features for RBP prediction between different species,

What is the implication of the main findings?

- Quick finding of candidate RBPs for further verification through biological experiments,
- Performance improvement of predicting RBP using computational methods,
- Recommendation for species-specific RBP predicting models.

Abstract: RNA-binding proteins (RBPs) play an important role in the synthesis and degradation of ribonucleic acid (RNA) molecules. The rapid and accurate identification of RBPs is essential for understanding the mechanisms of cell activity. Since identifying RBPs experimentally is expensive and time-consuming, computational methods have been explored to predict RBPs directly from protein sequences. In this paper, we developed an RBP prediction method named CnnRBP based on a convolution neural network. CnnRBP derived a sparse high-dimensional di- and tripeptide frequency feature vector from a protein sequence and then reduced this vector to a low-dimensional one using the Light Gradient Boosting Machine (LightGBM) algorithm. Then, the low-dimensional vectors derived from both RNA-binding proteins and non-RNA-binding proteins were fed to a multi-layer one-dimensional convolution network. Meanwhile, the SMOTE algorithm was used to alleviate the class imbalance in the training data. Extensive experiments showed that the proposed method can extract discriminative features to identify RBPs effectively. With 10-fold cross-validation on the training datasets, CnnRBP achieved AUC values of 99.98%, 99.69% and 96.72% for humans, *E. coli* and Salmonella, respectively. On the three independent datasets, CnnRBP achieved AUC values of 0.91, 0.96 and 0.91, outperforming the recent tripeptide-based method (i.e., TriPepSVM) by 8%, 4% and 5%, respectively. Compared with the state-of-the-art CNN-based predictor (i.e., iDRBP_MMC), CnnRBP achieved MCC values of 0.67, 0.68 and 0.73 with significant improvements by 6%, 6% and 15%, respectively. In addition, the cross-species testing shows that CnnRBP has a robust generalization performance for cross-species RBP prediction between close species.

Keywords: RNA-binding protein; convolution neural network; short peptide motifs; feature selection with LightGBM

Citation: Wei, Z.-S.; Rao, J.; Lin, Y.-J. A Deep Model for Species-Specific Prediction of Ribonucleic-Acid-Binding Protein with Short Motifs. *Appl. Sci.* **2023**, *13*, 8231. https://doi.org/10.3390/app13148231

Academic Editor: Alexander N. Pisarchik

Received: 31 May 2023
Revised: 5 July 2023
Accepted: 13 July 2023
Published: 15 July 2023

1. Introduction

RNA-binding proteins (RBPs) are a class of proteins that interact with RNA and regulate the metabolic process [1,2]. RBPs are involved in many biological processes such as gene regulation and apoptosis. They mediate the maturation, translocation, localization, modification, splicing and translation of RNAs through forming nucleic acid protein

complexes [3]. In the absence of RBPs, most RNAs cannot normally regulate and metabolize, except for a few RNAs that can act alone in the form of nuclease [4]. Moreover, RBPs may have one or more target RNAs, and their expression defects can cause a variety of diseases, such as neurodegenerative diseases, cancer and metabolic disorders [5,6]. The efficient identification of RBPs is important to medical research.

Traditional experimental methods are time-consuming and expensive because they identify RBPs via RNA interactome capture (RIC), which relies on in vivo UV cross-linking and oligo(dT) capture. Moreover, these methods may fail in some bacterial species that lack polyadenylation. The above reasons limit the application of traditional experimental methods to high-throughput data. The computational methods are time efficient, oligo(dT)-independent and can make up for the shortcomings of traditional experimental methods. Therefore, the computational RBP prediction methods have received more and more attention from researchers. The existing methods mainly describe proteins using features derived from protein primary sequences and then train classification models based on these features. The most commonly involved features include evolutionary conservation information, physicochemical properties, predicted structure information, position-specific scoring matrixes, physicochemical properties, and so on. And the most commonly used models include support vector machines, convolutional neural networks, and so on. However, recent studies on RBPs have made progress. Castello et al. [7] found that some short motifs (e.g., tripeptide or dipeptide) show specific conservation in the internal disordered regions (IDRs) of RBPs.

Although existing methods have made significant progress on RBP prediction, there is still a lot of room for improvement. Motivated by the work of Castello et al. [7], we propose a deep model named CnnRBP based on frequencies of the short motif of amino acid residues in this paper. The proposed method described proteins with short peptide motifs. A convolutional neural network was trained based on features derived from these short peptide motifs. Meanwhile, feature selection was carried out via the Light Gradient Boosting Machine (LightGBM) [8]. The performance of the proposed method was exhaustively evaluated through cross-validation and independent validation on the benchmark datasets. The rest of this paper is organized as follows: Section 2 describes the related works of RBP prediction. Section 3 presents the details of benchmark datasets, the proposed method and evaluation indexes. Section 4 presents the experimental results and makes an analysis. Finally, Section 5 makes conclusions.

2. Related Works

In the past decade, several computational methods have been proposed to predict RBPs. According to their feature representations to proteins, these methods can be roughly split into two categories, i.e., sequence-based and structure-based methods. Due to fact that the available structure data of proteins are relatively small, structure-based methods cannot be widely applied and have lacked attractiveness recently. To the best of our knowledge, the most recent work was the BindUP [9] proposed in 2016. It was a web server for the non-homology-based prediction of RBPs. BindUP extracted features, such as electrostatic surface patches, the molecular weight, surface accessibility and the moment dipole of the protein chain, from a 3D structure of a protein or a structural model. These numeric features were then fed to the supervised learning framework to train a support vector machine (SVM) classifier with a linear kernel function. In contrast, sequence-based methods are receiving more and more attention. Sequence-based methods derive various features from protein primary sequences, such as evolutionary conservation information, physicochemical properties or predicted structure information. Based on these features, they train classifiers to distinguish RBPs from non-RBPs via supervised learning. Position-specific scoring matrixes (PSSMs) and physicochemical properties are two kinds of features that are frequently utilized to predict RBPs. Kumar et al. [10] trained an SVM classifier denoted as RNAPred based on the PSSMs of proteins. RNAPred firstly identifies RBPs according to the number of predicted binding residues. If the number exceeds the threshold

for RBPs, the testing sample will be classified as an RBP. And if the number is less than the threshold for non-RBPs, the testing sample will be classified as a non-RBP. In other cases, the trained SVM is responsible for further prediction. The catRAPID [11] signature computed the Pearson correlation coefficient between physicochemical properties and annotated RNA-binding domains (RBDs) and trained an SVM with an RBF kernel based on the fraction of residues with a high correlation and associated RBDs. Similar to catRAPID, Sharan et al. [12] proposed the APRICOT method that tried to predict RBDs from protein sequences. They described RBD characteristics using PSSMs, instead of physicochemical properties. Zhang and Liu integrated physicochemical properties and PSSMs to train an SVM model named RBPPred [13]. They demonstrated a performance improvement for the integrated features over any single feature. Besides PSSMs and physicochemical properties, the structure features predicted from sequences are also used to identify RBPs. For example, SPOT-Seq-RNA [14] combined template-based structure-prediction software with binding-affinity prediction software for protein–RNA complexes. Recently, Bressin et al. [15] introduced a novel sequence-derived feature to train an SVM model, TriPepSVM, to predict RBPs. They computed frequencies of tripeptide motifs by continuously moving a sliding window in a protein sequence. On the dataset of three species, experimental results demonstrated that this feature was computationally efficient and discriminative for identifying RBPs.

With the wide application of deep learning in bioinformatics fields [16–20], several works tried to improve the performance of RPB predictors by resorting to deep learning. Deep-RBPPred [21] combined five physicochemical properties using a global composition feature encoding method (CTD) to form a 160-dimension feature vector and then train a convolutional neural network (CNN) with the reshaped 8 × 20 feature tensor as input. Furthermore, DeepMVF-RBP [22] encoded physicochemical properties using three different methods, i.e., CTD, conjoint triad (CT) and parallel correlation pseudo amino acid composition (PC-PseAAC) and then concatenated them together to train a deep belief network (DBN). And Zhao and Du [23] clustered amino acids into seven groups according to physicochemical properties and trained a CNN based on one-hot encoding and a CNN with a residual block based on conjoint triad encoding. Then, the two CNNs made up an ensemble classifier named econvRBP. Zhang et al. [24] treated the prediction of RBPs and DBPs as a multi-label classification problem. Using multi-label learning methods, they developed a CNN model named iDRBP_MMC based on PSSMs and known motifs to predict RBPs and DNA-binding proteins (DBPs) simultaneously. In addition, Pan et al. [25] proposed an LSTM model to identify binding proteins for RNA sequences using multi-label deep learning. And Niu et al. [26] identified binding proteins for RNA sequences via a deep learning model with a CNN layer followed by an LSTM layer.

Although deep learning methods have achieved an exhilarating performance for RBP prediction, there are still some disadvantages. For example, Deep-RBPPred inputs simple physicochemical properties and can make decisions quite efficiently but shows a moderate performance [21], whereas iDRBP_MCC achieves a state-of-the-art performance but is time-consuming since it inputs PSSMs obtained through sequence alignments on a large protein database [24]. Meanwhile, iDRBP_MCC has to perform zero padding when the length of the input protein is different from the preset value, which may lead to a decline in the case that the length of the test data is distinct from that of the training data.

3. Materials and Methods

The flowchart of CnnRBP is shown in Figure 1. Firstly, the protein sequence is described by a feature vector of the frequencies of di- and tripeptide motifs. The frequencies of di- and tripeptide motifs are calculated directly from protein sequences, which is both a high-performance method and computationally efficient. The dimension of the feature vector is constant despite different protein sequence lengths, and therefore zero padding is avoided. Secondly, LightGBM [8] was utilized to select important features from the initial sparse high-dimensional feature vector and reduce the dimension of the feature vector.

Lastly, a convolutional neural network (CNN) with one-dimension convolutional filters captured more sophisticated relations among the low-dimensional features and made the final prediction. In addition, to alleviate the harmful effects of class imbalance existing in training datasets, the Synthetic Minority Oversampling Technique (SMOTE) [27] was introduced to increase the size of the minority class (positive) data. Based on the proposed method, three species-specific RBP predictors were trained on the datasets of three species, humans, E. coli and Salmonella, respectively. The ten-fold cross-validation on training datasets and the independent validation on testing datasets were carried out to evaluate the proposed predictors and compare them with existing methods. The cross-species evaluations were also adopted to show the generalization performance of the proposed CnnRBP for cross-species prediction.

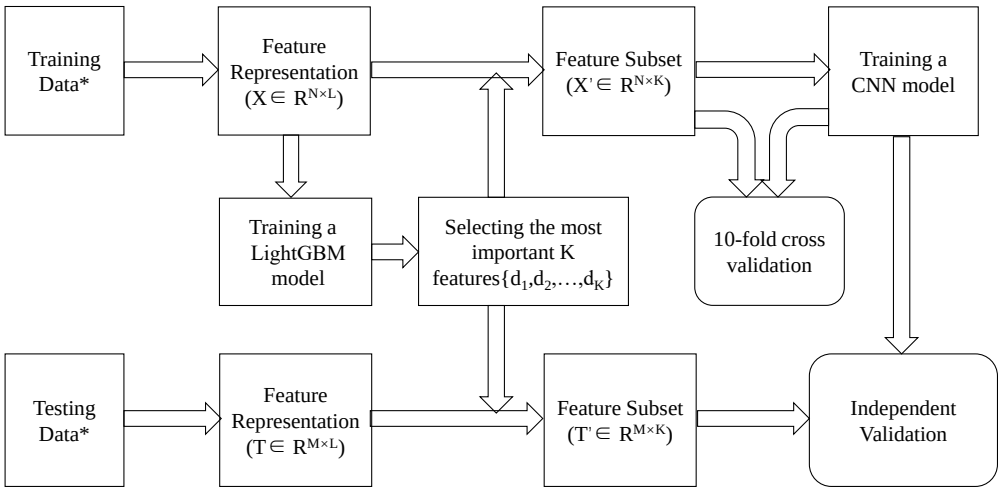


Figure 1. The flowchart of the proposed CnnRBP. * For each species, one model is trained on its training dataset and then tested on its testing dataset.

3.1. Benchmark Datasets

To evaluate the performance of the proposed CnnRBP, six publicly available datasets [15] were utilized as benchmark datasets. These datasets contain RNA-binding and non-RNA-binding proteins from three different species, humans, *E. coli* and Aalmonella, with two datasets per species. For each species, one dataset was taken as the training dataset and the other one was utilized to perform independent validation. For humans, the training dataset comprises 1625 positive samples (RNA-binding proteins) and 10,834 negative samples (non-RNA-binding proteins), and the independent validation dataset comprises 181 positive samples and 1204 negative samples. For *E. coli*, the training dataset comprises 460 positive samples and 3404 negative samples, and the independent validation dataset comprises 52 positive samples and 379 negative samples. For Salmonella, the training dataset comprises 275 positive samples and 1273 negative samples, and the independent validation dataset comprises 31 positive samples and 142 negative samples. Table 1 summarizes the statistics of these above six benchmark datasets. Please refer to the paper [15] for details of these datasets.

Table 1. Sizes of training and independent validation datasets for all three species.

Dataset	Training			Independent Validation		
	Positive	**Negative**	**Pos:Neg**	**Positive**	**Negative**	**Pos:Neg**
Human	1625	10,834	1:6.67	181	1204	1:6.65
E. coli	460	3404	1:7.4	52	379	1:7.29
Salmonella	275	1273	1:4.63	31	142	1:4.58

3.2. Feature Representation Based on Short Peptide Motifs

Sequence motifs are short amino acid composite patterns that are conserved during protein evolution. Several di- and tripeptides have been found to occur more frequently in the structural disorders of RBPs [7]. Therefore, we described protein sequences based on di- and tripeptide motifs. In details, a protein was described as a vector of the frequencies for di- and tripeptides calculated from the amino acid sequence. As shown in Figure 2, we moved a sliding window of a 2- and 3-amino-acid size on the protein sequence one amino acid at a time, and counted the occurrence frequencies of all possible di- and tripeptides. For example, given a protein sequence of 'TYSYHKYSYT', the dipeptide motif of 'YS' appears twice when moving a sliding window of size 2 from left to right, then the place in the feature vector counting the frequency of the 'YS' motif is set to 2. There are 400 dipeptides and 8000 tripeptides in all, so we obtain an 8400 D feature vector from a protein sequence.

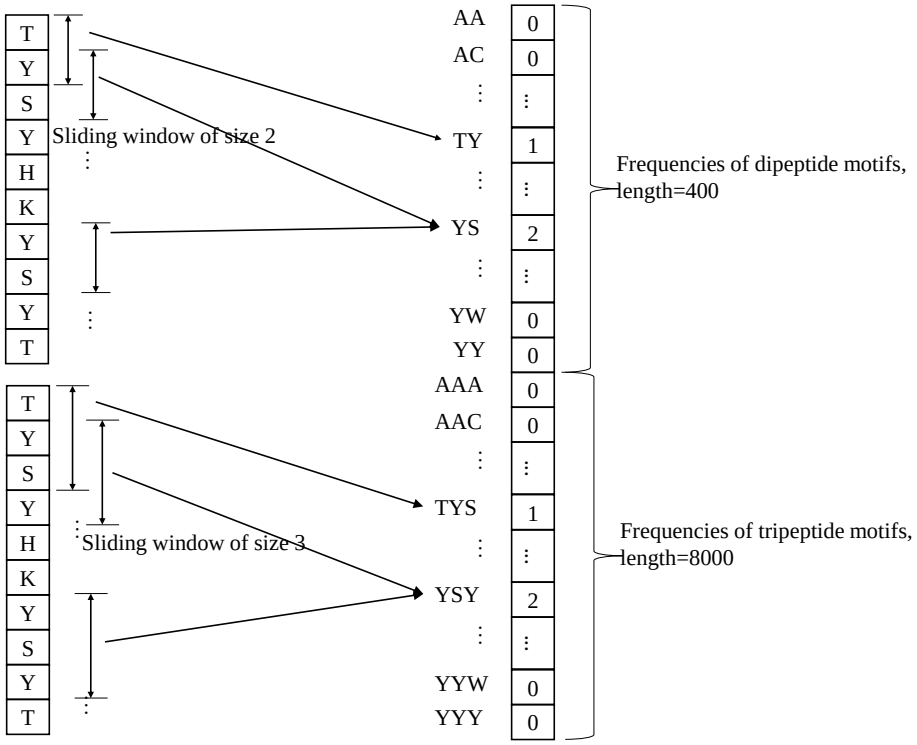


Figure 2. Feature representation based on di- and tripeptide motifs.

3.3. Feature Selection Using the LightGBM Algorithm

Our feature vectors for proteins are sparse high-dimensional vectors, which may lead to a negative impact on the training of subsequent CNN models. To improve the performance of subsequent classification models, it is common to apply feature selection to the initial feature vectors ahead of the training the classification model. Therefore, we carried out feature selection by estimating feature importance based on the LightGBM algorithm [8]. Specifically, we firstly trained a LightGBM model with initial feature vectors on the training dataset and then calculated the feature importance as the average importance over all decision trees of the trained LightGBM model, where the importance of a feature for a decision tree is measured by the sum of the decrease in gini impurity over the internal nodes that choose the variable (feature) to partition the associated region into two subregions. The higher the score of the feature, the higher the importance of the feature. Then, all features are sorted by their gain scores. Finally, the optimal feature subset is

selected according to the sorted features. As a result, the importance of a feature ℓ can be calculated as follows:

$$\mathcal{I}_\ell = \frac{1}{M}\sum_{m=1}^{M}\mathcal{I}_\ell(T_m) \tag{1}$$

$$\mathcal{I}_\ell(T) = \sum_{t=1}^{J}\triangle\mathcal{G}_t I(v(t)=\ell) \tag{2}$$

where, $\mathcal{I}_\ell(T)$ represents the importance of the feature ℓ for the tree T, $\triangle\mathcal{G}_t$ represents the decrease in gini impurity in the internal node t, $I()$ is the indicator function, $v(t)$ represents the chosen variable (feature) for the node t, M is the number of trees in the LightGBM model and J is the number of internal nodes in the decision tree T.

After feature selection, we use Formula (3) to normalize each feature to a real number between 0 and 1, where x is the original value of each feature and max and min are the maximum and minimum values of each feature in the training dataset. In this way, we eliminated differences in the magnitude for all features and ensured that the convolutional neural network can correctly capture the potential relationship between features.

$$x^* = \frac{x - min}{max - min} \tag{3}$$

3.4. CnnRBP Model

Recently, deep learning has been applied in several problems in the bioinformatics field and achieved an excellent performance. In this paper,we proposed an RNA-binding protein prediction model based on a convolution neural network (CNN), as shown in Figure 3. The CNN has achieved great success in image recognition, image segmentation, video recognition, pattern recognition, natural language processing and other fields. Inspired by the success of the CNN in the feature extraction of two-dimensional image data, we applied a one-dimensional CNN to extract local features from a protein sequence to capture the relations between adjacent residues.

As shown in Figure 3, our model contains a CNN framework with five one-dimensional convolutional layers. Each convolutional layer contains 128 convolutional kernels with different sizes (5, 5, 3, 3 and 1, respectively). These convolutional layers are expected to extract sophisticated features from sequence motifs. Meanwhile, to make the extracted features more robust and reduce the size of the representations, there is a max pooling layer with a window of 3 and a step size of 2 after each of the first 4 convolutional layers and a max pooling layer with a window of 3 and a step size of 1 after the last convolutional layer. For feature mapping, the relu function (4) with a small influence function kernel is used as the activation function of each convolutional neuron.

$$relu(x) = max(0, x) \tag{4}$$

The extracted feature vector from the convolutional layer is then fed into a full connection framework of 3 layers to learn relations between features. To prevent over-fitting, a dropout layer with a parameter of 0.75 is added between two adjacent full connection layers. In the output layer, two neurons output the probabilities of being an RBP and a non-RBP, respectively, calculated by the softmax activation function. To train this network, we use the standard cross-entropy as the loss function, which is shown in Formula (5), to minimize the training error.

$$Loss = \frac{1}{N}\sum -[y_i \log(p_i) + (1 - y_i)\log(1 - p_i)] \tag{5}$$

where N is the total number of training samples, y_i is the label of the *i-th* sample, the positive class is 1, the negative class is 0 and p_i is the probability that the *i-th* sample is predicted to be positive.

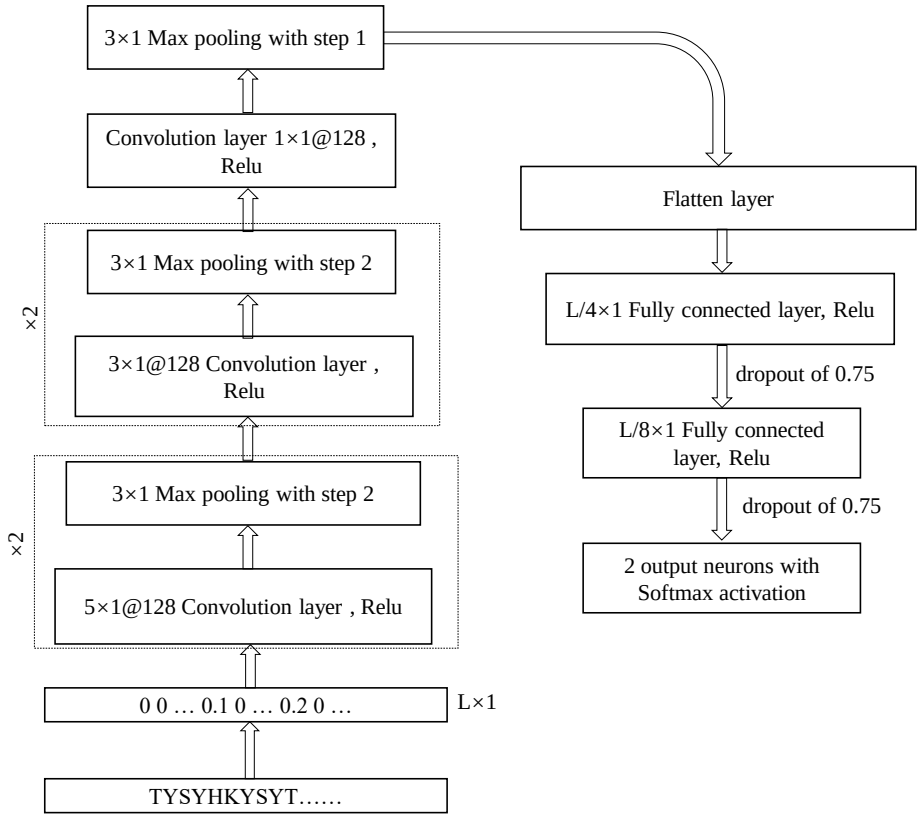


Figure 3. The framework of the CnnRBP model. L is the length of the feature subset selected by the LightGBM algorithm.

Because the data are highly unbalanced, we use the SMOTE algorithm [27] to alleviate the bias problem in model prediction. Specifically, we apply the SMOTE algorithm on the feature vector space of the positive class to make the sample number of the positive class almost equal to that of the negative class. Then, based on the balanced dataset, the feature selection and training of the CNN model are carried out.

3.5. Performance Evaluation

To evaluate the prediction performance of the proposed model and compare it with other models, eight evaluation indexes are measured on training datasets and independent datasets. These indexes include the accuracy (ACC), precision (PRE), sensitivity (SEN), specificity (SPE), F1 measure (F1), Matthew's correlation coefficient (MCC), balanced accuracy (BACC), area under precision recall curve (AUPR) and area characteristic curve under the prediction receiver operator (AUC), which are defined as follows:

$$ACC = \frac{TP + TN}{TP + TN + FP + FN}. \tag{6}$$

$$PRE = \frac{TP}{TP + FP}. \tag{7}$$

$$SEN = \frac{TP}{TP + FN}. \tag{8}$$

$$SPE = \frac{TN}{TN + FP}. \tag{9}$$

$$F1 = \frac{2 \times SEN \times PRE}{SEN + PRE}. \quad (10)$$

$$MCC = \frac{TP \times TN - FP \times FN}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}}. \quad (11)$$

$$BACC = \frac{(\frac{TP}{P} + \frac{TN}{N})}{2}. \quad (12)$$

where P is the number of positive samples, N is the number of negative samples, TP is the number of positive samples that are correctly predicted, FP is the number of negative samples that are incorrectly predicted, TN is the number of negative samples that are correctly predicted and FN is the number of positive samples that are incorrectly predicted.

Considering the imbalance in testing datasets, the ACC cannot accurately evaluate the performance of classification models. Alternatively, the F1 measure, MCC and BACC can evaluate the overall performance of a model more appropriately. In addition, the ROC curve and PR curve are frequently used to assess the overall performance of binary classification models, so the AUC and AUPR are calculated to compare the performance of classification models more directly.

4. Experimental Results and Analysis

4.1. Performance on Training Datasets through Cross-Validation

The CNN in the proposed CnnRBP model contains several hyperparameters. To tune these hyperparameters to achieve the best prediction performance, we carried out 10-fold cross-validation with different hyperparameters on the training datasets for three species separately. The details are described as follows:

Firstly, the training dataset is divided into 10 independent subsets, in which the proportion of the positive to the negative is the same as that in the whole dataset. Then, one subset is taken as the test set, and the remaining nine subsets are taken together as the training set. This procedure is carried out 10 times to make each subset the test set once. Finally, for each validation, we use the SMOTE algorithm to balance the training set to train a CnnRBP model and then perform predictions on the test set with the trained model. The averaged prediction results for all 10-time validations are reported.

By means of 10-fold cross-validation, we empirically explore several hyperparameters including the number of filters in the convolution layer ($filters$ = 32, 64, 128), kernel size (k = 1, 3, 5), optimizer and learning rate (lr = 0.001, 0.002, 0.003, 0.004). As a result, the number of filters is set to 128. The kernel sizes of the five convolution layers are set to 5, 5, 3, 3 and 1, respectively. The learning rate is set to 0.001, and the Adam optimizer is selected as the optimizer. The prediction results with the above settings are shown in Table 2 through 10-fold cross-validation of the training datasets.

Table 2. Performance on training datasets through 10-fold cross-validation.

Dataset	ACC (%)	PRE (%)	SEN (%)	SPE (%)	F1 (%)	BACC (%)	MCC (%)	AUC (%)
Human	99.91	99.96	99.78	99.98	99.87	99.88	99.81	99.98
E. coli	99.08	99.59	97.65	99.79	98.57	98.72	97.95	99.69
Salmonella	95.34	94.09	92.34	96.86	92.85	94.60	89.79	96.72

As shown in Table 2, our models achieve AUC values over 96% on the three species datasets. More specifically, our models achieve ACC, MCC and AUC values on the human dataset of 99.91%, 99.81% and 99.98%, respectively. On the *E. coli* dataset, the values of the ACC, MCC and AUC are 99.08%, 97.95% and 99.69%, respectively; On the Salmonella dataset, the values of the ACC, MCC and AUC are 95.34%, 89.79% and 96.72%, respectively. In addition, our models achieve BACC values of 99.88%, 98.72% and 94.60% on the three species datasets, respectively. As a comparison, TriPepSVM reports BACC values of 75.7%

on the human dataset, 84.1% on the *E. coli* dataset and 82.7% on the Salmonella dataset through 10-fold cross-validation in their supplementary materials. This shows that the proposed method can achieve a great overall performance on unbalanced datasets.

4.2. Performance Analysis with Different Feature Dimensionalities

Another key parameter is the feature dimensionality for feature selection via the LightGBM algorithm. To find out the optimal value of the feature dimensionality, we explored the AUC values with different dimensionalities through 10-fold cross-validation. Firstly, we set the feature dimensionalities as 50, 500, 1000, 2000, 3000 and 5000, respectively, and found that the optimal value should be between 1000 and 2000. Then, we varied the values from 1000 to 2000 with a step size of 100 and recorded AUC values in all cases. At last, the AUC values versus the feature dimensionalities are plot in Figure 4 through 10-fold cross-validation on the training datasets of three species.

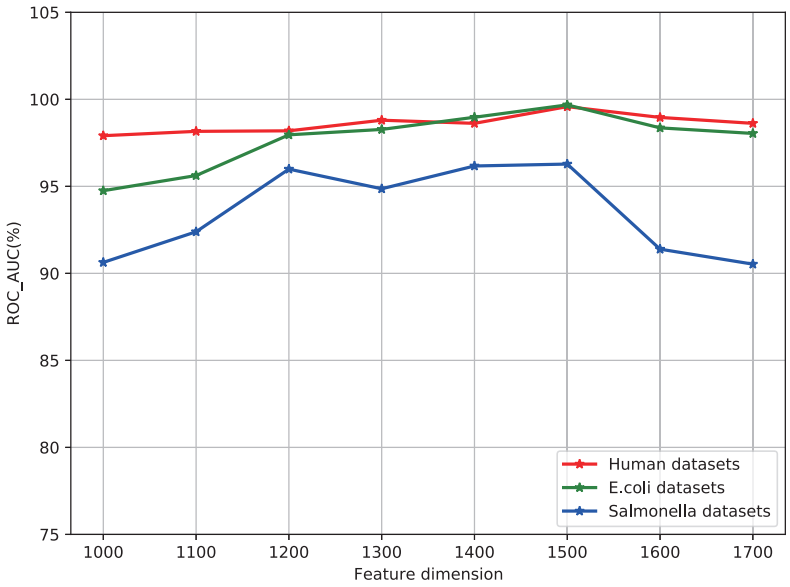


Figure 4. AUC values of different species datasets under different feature dimensions.

As shown in Figure 4, the AUC values fluctuate similarly with different feature dimensionalities on the three training datasets. Specifically, the optimal value is 1500 for feature dimensionalities on the datasets of the three species. When the feature dimensionality is less than 1500, the AUC values mainly increase with increases in the values of feature dimensionality. On the contrary, the AUC values gradually decrease with feature dimensionality over 1500.

To further explore the correlation between different species, we examined the selected features. We found that the overlapping rate of selected features between human and *E. coli* species was 28.9%. The overlapping rate of selected features between the human and Salmonella species was also only 30.7%. In contrast, the overlapping rate of selected features between Salmonella and *E. coli* species reached 45.1%. Table 3 summarizes the top 10 and last 10 selected features for different species. It can be found that humasn and *E. coli* species shared 2 features in the top 10 features. Human and Salmonella species also only shared 3 features in the top 10 features, while *E. coli* and Salmonella species shared 5 features. However, no two species shared features in the last 10 features. The above results indicate that RBPs of different species have different characteristics. Meanwhile, the RBPs of closer species have more similar characteristics. This also explains the difference in cross-species prediction performance in Section 4.4.

Table 3. The top and last selected features for different species.

The Top 10 Features										
Human	KK	SS	SL	LLL	CC	GK	GRG	FF	LL	RV
E. coli	RV	RK	KG	RL	AL	KR	SL	PF	RR	RTK
Salmonella	RK	KV	AL	RR	RV	TL	KR	LL	IK	KK
The Last 10 Features										
Human	DAQ	DAW	DVV	DAD	DAE	DAK	ACQ	DAR	DAH	DAP
E. coli	NHI	NHL	DLN	CSV	NRP	NRI	NRL	NRV	NRA	NRG
Salmonella	DGR	DGK	FW	DGE	DGD	DGQ	DGN	DGM	QHL	TF

4.3. Comparison with Other RBP Prediction Methods on Independent Validation Datasets

To further evaluate the performance of the proposed CnnRBP method, we compare our method with existing RBP prediction methods, including SPOT-Seq-RNA [14], RNAPred [10], RBPPred [13], Deep-RBPPred [21], TriPepSVM [15] and iDRBP_MMC [24]. Among the six methods, TriPepSVM is an SVM model based on the same short sequence motifs as ours. Deep-RBPPred is a CNN model with features derived from five physicochemical properties. iDRBP_MMC is a multi-label learning model with a CNN based on PSSMs and structural motifs. For each species, we train a CnnRBP model on the training dataset with the hyperparameters optimized by the 10-fold cross-validation and then perform predictions on the independent validation dataset. For SPOT-Seq-RNA, RNAPred, RBPPred and TriPepSVM, their evaluation data are cited from the paper that proposed the method TriPepSVM. For Deep-RBPPred, we executed the publicly released source code to make an evaluation on the independent validation dataset. For iDRBP_MMC, we fed testing samples into its web server and then calculated evaluation indexes from the prediction results. The evaluation indexes for all methods are shown in Table 4.

Table 4. Comparison with existing RBP prediction methods on independent validation datasets.

Predictor	ACC (%)	PRE (%)	SEN (%)	SPE (%)	F1 (%)	BACC (%)	MCC	AUC
Human dataset								
SPOT-Seq-Pred	85.70	38.89	23.20	94.52	29.07	58.86	0.22	-
RNAPred	49.67	18.74	87.57	44.09	30.88	65.83	0.22	0.72
RBPPred	65.63	24.08	75.69	64.12	36.53	69.91	0.27	0.70
Deep-RBPPred	30.25	14.73	90.61	21.18	25.35	55.90	0.10	0.69
TriPepSVM	88.95	58.75	51.93	94.45	55.13	73.23	0.49	0.83
iDRBP_MMC	89.29	56.57	78.45	90.93	65.73	84.69	0.61	0.92
CnnRBP *	92.64	72.07	71.27	95.85	70.67	83.56	0.67	0.91
E. coli dataset								
SPOT-Seq-Pred	87.28	100.00	29.03	100.00	45.00	64.52	0.50	-
RNAPred	66.67	32.00	80.00	63.83	45.71	71.91	0.34	0.75
RBPPred	80.92	47.73	67.74	83.80	56.00	75.77	0.45	0.77
Deep-RBPPred	61.02	20.71	78.85	58.58	32.80	68.72	0.24	0.72
TriPepSVM	92.34	69.39	65.38	96.04	67.32	80.71	0.63	0.92
iDRBP_MMC	92.57	75.00	57.00	97.36	64.77	77.18	0.62	0.90
CnnRBP *	93.04	69.64	75.00	95.51	72.22	85.26	0.68	0.96
Salmonella dataset								
SPOT-Seq-Pred	92.11	100.00	34.15	100.00	51.43	67.31	0.56	-
RNAPred	49.18	17.25	86.27	44.18	28.76	65.23	0.20	0.79
RBPPred	81.44	35.11	63.46	83.91	45.21	73.68	0.37	0.81
Deep-RBPPred	60.12	27.38	74.19	57.04	40.00	65.62	0.24	0.70
TriPepSVM	90.17	85.00	54.83	97.89	66.66	76.36	0.63	0.86
iDRBP_MMC	87.86	67.85	61.29	93.66	64.41	77.48	0.58	0.90
CnnRBP *	92.49	84.62	70.97	97.18	77.19	84.08	0.73	0.91

* Training one model per species.

From the results in Table 4, it can be found that our method CnnRBP achieves the best overall prediction performance among all methods. Specifically, on the human dataset we can find that two deep-learning-based methods (i.e., CnnRBP and iDRBP_MMC) are significantly superior compared to other shallow methods, according to the overall performance indexes, such as the F1 measure, BACC, MCC and AUC. Among the shallow methods, TriPepSVM performs best with an AUC of 0.83, MCC of 0.49 and BACC of 73.23%. RNAPred and SPOT-Seq-Pred appear to have a large disparity in SEN and SPE, leading to a poor performance in BACC and MCC, which shows that they cannot deal with the class imbalance problem. In the three deep-learning-based methods, our method CnnRBP performs slightly worse than iDRBP_MMC in the AUC and BACC by about 1%. However, CnnRBP achieves a significantly better performance than iDRBP_MMC in the other two overall indexes of the F1 measure and MCC with improvements of 4.9% and 6%. Meanwhile, CnnRBP significantly outperforms iDRBP_MMC in the ACC, PRE and SPE. It is worth noting that CnnRBP only makes use of statistical information directly from amino acid sequences, while iDRBP_MMC takes as input more complicated PSSMs obtained via sequence alignments. Another deep-learning-based method (i.e., Deep-RBPPred) performs poorly, maybe due to the feature representation being too simple.

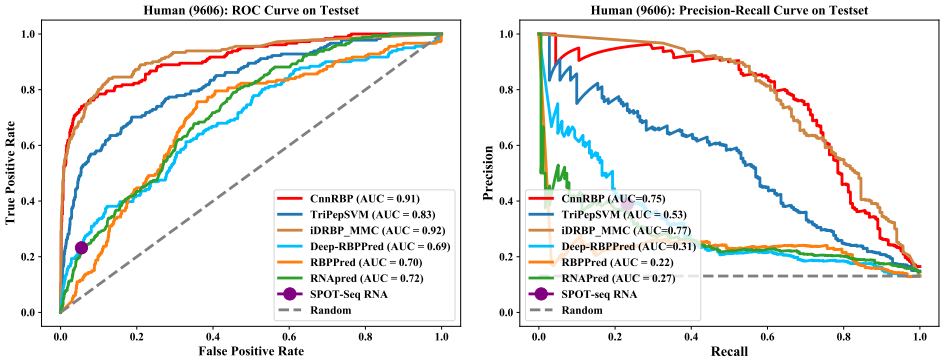
On the *E. coli* dataset, our method performs best among all methods with significant improvements in the ACC and all overall indexes, such as the F1 measure, BACC, MCC and AUC. Compared with another deep model, iDRBP_MMC, our method achieves improvements of 6%, 6% and 8.08% in the AUC, MCC and BACC, respectively. It can be found that our method performs slightly worse than iDRBP_MCC in SPE but achieves a significantly better performance in SEN, which shows that our method has advantages over iDRBP_MMC in identifying RBPs. On this dataset, the shallow method TriPepSVM performs similarly or even better than iDRBP_MMC, which shows the effectiveness of short motifs as protein representation for RBP detection.

On the Salmonella dataset, our method performs best in almost all indexes. Specifically, our method achieves an AUC of 0.91, MCC of 0.73, F1 measure of 77.19% and BACC of 84.08%, respectively. Compared with iDRBP_MMC, our method makes improvements in all indexes. Our method outperforms iDRBP_MMC with improvements of 12.78%, 6.6%, 15% and 1% in the F1 measure, BACC, MCC, and AUC, respectively. Compared with TriPepSVM, our method achieves significant improvements in the F1 measure, BACC, MCC and AUC by 10.53%, 7.72%, 10% and 5%, respectively. These results show that the prediction performance of our method is better than other existing methods with the help of the deep learning framework.

To compare the performances of different methods more intuitively, the ROC curves and PR curves are drawn in Figures 5–7 for the compared methods on the independent validation datasets of the three species.

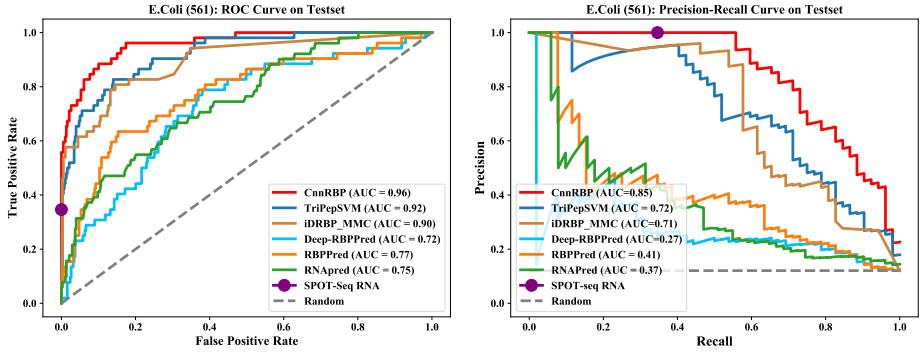
As shown in Figure 5, the two deep models, i.e., CnnRBP and iDRBP_MMC, are close to each other for both the ROC curves and PR curves on the human dataset, significantly outperforming other shallow models. Furthermore, based on the same features, the proposed CnnRBP is superior to TriPepSVM, which shows that the deep model can achieve a better performance than the shallow model. Deep-RBPPred and the other three shallow models show poor performances on this dataset, probably due to weak feature representations.

As shown in Figures 6 and 7, a similar conclusion to the one above can be found, which is that the deep models outperform the shallow models. Specifically, our method outperforms another deep model (i.e., iDRBP_MMC) significantly on these two datasets, indicating the advantages of species-specific models. Our method trained species-specific models for different species, which took into account the diversity of species.



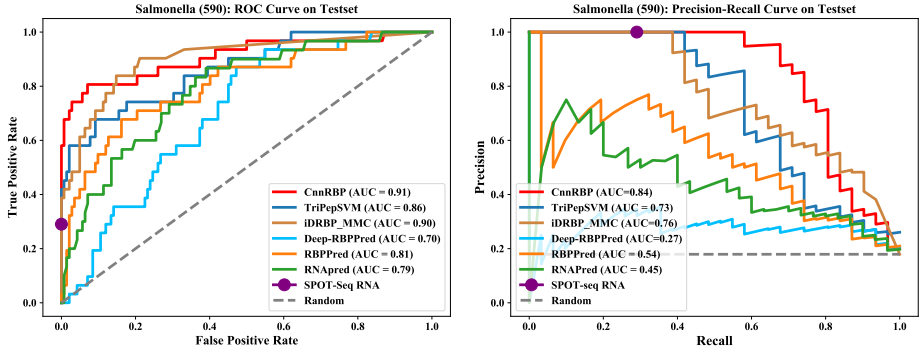
(**a**) ROC curves (**b**) PR curves

Figure 5. Comparison of AUPR and AUC values of the proposed CnnRBP with other methods on human independent validation dataset.



(**a**) ROC curves (**b**) PR curves

Figure 6. Comparison of AUPR and AUC values of the proposed CnnRBP with other methods on *E. coli* independent validation dataset.



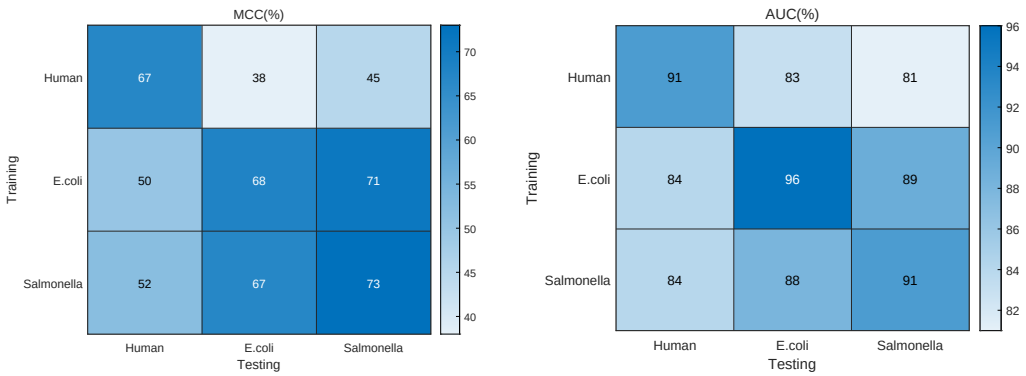
(**a**) ROC curves (**b**) PR curves

Figure 7. Comparison of AUPR and AUC values of the proposed CnnRBP with other methods on Salmonella independent validation dataset.

In conclusion, benefiting from the strong representation ability of deep learning, our method achieved a significantly greater performance improvement compared to the shallow models. Meanwhile, our method was quite competitive when compared to the other deep model, despite using simpler features.

4.4. Cross-Species Prediction

In this section, we explore the cross-species performance of our method experimentally. We trained a model on the training dataset of one species and then evaluated it on the independent test datasets of all three species. The results are shown in Figure 8. It can be found that the models trained on *E. coli* and Salmonella datasets achieved MCC values of 0.50 and 0.52, respectively, when testing on the human test dataset, which shows a much worse performance than that trained on the human dataset with an MCC value of 0.67. Moreover, when testing on *E. coli* and Salmonella testing datasets, the model trained on the human dataset performed poorly with MCC values of 0.38 and 0.45, respectively. However, the models trained on the *E. coli* and Salmonella datasets had similar performances on all testing datasets of the three species. The AUCs of cross-species prediction also show that the models trained on the *E. coli* and Salmonella datasets had closer AUCs. It is worth noting that the model trained on *E. coli* had a slightly higher MCC on Salmonella than *E. coli*. However, the model trained on *E. coli* still had a significantly higher AUC on *E. coli* than Salmonella. These results demonstrate that the proposed CnnRBP method has a robust performance for cross-species prediction between close species, i.e., *E. coli* and Salmonella. When two species are quite different, the performance of the model decrease significantly, which suggests the necessity of a species-specific prediction model.



(**a**) (**b**)

Figure 8. Performance on cross-species prediction: (**a**) MCC; (**b**) AUC.

5. Conclusions

The accurate prediction of RBPs based on sequences has been a challenging problem. In this paper, we have proposed a deep model (i.e., CnnRBP) based on short sequence motifs to improve the performance of RBP prediction. Cross-validation and independent validation tests on the datasets of three species demonstrated the effectiveness of our method and its superiority over existing shallow and deep learning based methods. The short motifs are easily computed and discriminative, as shown by experimental results. The LightGBM algorithm was utilized to select important features and the convolution neural network found relations among features, which contributes to performance improvements for our RBP prediction method. Our method can help to quickly find candidate RBPs for further verification through biological experiments.

In summary, experimental results on the benchmark datasets show that our method achieved a fairly competitive prediction performance. Compared with other deep models, the features used by our method were computationally efficient. In addition, species-specific predictions can capture unique features of different species. However, our method still has some disadvantages. When the protein is too short, the statistical features may be too sparse to make correct decisions. The CNN captures local features but cannot find out the context information. The contributions are summarized below: We proposed a novel species-specific RBP predictor. We also verified that the discriminant features for

RBP prediction differed between species. In addition, we introduced LightGBM to select optimal features.

Although deep learning has been applied to improve the performance of species-specific RBP prediction in this paper, there is still room for further improvement. In some species, such as E.coli and Salmonella, the number of RBP samples is relatively small, which limits the advantages of deep learning. Next, we will further explore transfer learning to improve the performance for species with small RBP samples by transferring from the models trained on the datasets of species with large RBP samples.

Author Contributions: Conceptualization, Z.-S.W. and J.R.; data curation, J.R.; formal analysis, Z.-S.W.; funding acquisition, Y.-J.L.; investigation, Z.-S.W.; methodology, Z.-S.W. and J.R.; project administration, Z.-S.W.; software, J.R.; supervision, Z.-S.W.; validation, Z.-S.W.; visualization, Z.-S.W. and J.R.; writing—original draft, Z.-S.W.; writing—review and editing, Z.-S.W. and Y.-J.L. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported by the National Natural Science Foundation of China (No. 62076116), the Natural Science Foundation of Fujian (No. 2022J01913) and the Education and Research Project for Young and Middle-aged Teachers of the Education Department of Fujian Province (No. JAT190362).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The datasets used are available publicly.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Lu, Z.; Guan, X.; Schmidt, C.A.; Matera, A.G. RIP-seq analysis of eukaryotic Sm proteins identifies three major categories of Sm-containing ribonucleoproteins. *Genome Biol.* **2014**, *15*, R7. [CrossRef] [PubMed]
2. Marchese, D.; de Groot, N.S.; Lorenzo Gotor, N.; Livi, C.M.; Tartaglia, G.G. Advances in the characterization of RNA-binding proteins. *WIREs RNA* **2016**, *7*, 793–810. [CrossRef] [PubMed]
3. Xiao, R.; Chen, J.Y.; Liang, Z.; Luo, D.; Fu, X.D. Pervasive Chromatin-RNA Binding Protein Interactions Enable RNA-Based Regulation of Transcription. *Cell* **2019**, *178*, 107–121. [CrossRef]
4. Fei, T.; Chen, Y.; Xiao, T.; Li, W.; Cato, L.; Zhang, P.; Cotter, M.B.; Bowden, M.; Lis, R.T.; Zhao, S.G.A. Genome-wide CRISPR screen identifies HNRNPL as a prostate cancer dependency regulating RNA splicing. *Proc. Natl. Acad. Sci. USA* **2017**, *114*, E5207–E5215. [CrossRef]
5. Gerstberger, S.; Hafner, M.; Tuschl, T. A census of human RNA-binding proteins. *Nat. Rev. Genet.* **2014**, *15*, 829–845. [CrossRef] [PubMed]
6. Hentze, M.W.; Castello, A.; Schwarzl, T.; Preiss, T. A brave new world of RNA-binding proteins. *Nat. Rev. Mol. Cell Biol.* **2018**, *19*, 327–341. [CrossRef]
7. Castello, A.; Fischer, B.; Frese, C.; Horos, R.; Alleaume, A.M.; Foehr, S.; Curk, T.; Krijgsveld, J.; Hentze, M. Comprehensive Identification of RNA-Binding Domains in Human Cells. *Mol. Cell* **2016**, *63*, 696–710. [CrossRef]
8. Ke, G.; Meng, Q.; Finley, T.; Wang, T.; Chen, W.; Ma, W.; Ye, Q.; Liu, T.Y. LightGBM: A Highly Efficient Gradient Boosting Decision Tree. In *Advances in Neural Information Processing Systems*; Curran Associates, Inc.: Red Hook, NY, USA, 2017; Volume 30, pp. 3146–3154.
9. Paz, I.; Kligun, E.; Bengad, B.; Mandel-Gutfreund, Y. BindUP: A web server for non-homology-based prediction of DNA and RNA binding proteins. *Nucleic Acids Res.* **2016**, *44*, W568–W574. [CrossRef]
10. Kumar, M.; Gromiha, M.M.; Raghava, G.P.S. SVM based prediction of RNA-binding proteins using binding residues and evolutionary information. *J. Mol. Recognit.* **2011**, *24*, 303–313. [CrossRef]
11. Livi, C.M.; Klus, P.; Delli Ponti, R.; Tartaglia, G.G. catRAPID signature: Identification of ribonucleoproteins and RNA-binding regions. *Bioinformatics* **2016**, *32*, 773–775. [CrossRef]
12. Sharan, M.; Förstner, K.U.; Eulalio, A.; Vogel, J. APRICOT: An integrated computational pipeline for the sequence-based identification and characterization of RNA-binding proteins. *Nucleic Acids Res.* **2017**, *45*, e96. [CrossRef] [PubMed]
13. Zhang, X.; Liu, S. RBPPred: Predicting RNA-binding proteins from sequence using SVM. *Bioinformatics* **2017**, *33*, 854–862. [CrossRef]
14. Yang, Y.; Zhao, H.; Wang, J.; Zhou, Y. SPOT-Seq-RNA: Predicting Protein-RNA Complex Structure and RNA-Binding Function by Fold Recognition and Binding Affinity Prediction. *Methods Mol. Biol.* **2014**, *1137*, 119–130. [PubMed]
15. Bressin, A.; Schulte-Sasse, R.; Figini, D.; Urdaneta, E.C.; Beckmann, B.M.; Marsico, A. TriPepSVM: De novo prediction of RNA-binding proteins based on short amino acid motifs. *Nucleic Acids Res.* **2019**, *47*, 4406–4417. [CrossRef] [PubMed]

16. Pouyanfar, S.; Sadiq, S.; Yan, Y.; Tian, H.; Tao, Y.; Reyes, M.P.; Shyu, M.L.; Chen, S.C.; Iyengar, S.S. A Survey on Deep Learning: Algorithms, Techniques, and Applications. *ACM Comput. Surv.* **2019**, *51*, 1–36. [CrossRef]
17. Ahmed, S.; Kabir, M.; Arif, M.; Khan, Z.U.; Yu, D.J. DeepPPSite: A deep learning-based model for analysis and prediction of phosphorylation sites using efficient sequence information. *Anal. Biochem.* **2021**, *612*, 113955. [CrossRef]
18. Hu, J.; Zheng, L.L.; Bai, Y.S.; Zhang, K.W.; Yu, D.J.; Zhang, G.J. Accurate prediction of protein-ATP binding residues using position-specific frequency matrix. *Anal. Biochem.* **2021**, *626*, 114241. [CrossRef]
19. He, W.; Wang, Y.; Cui, L.; Su, R.; Wei, L. Learning embedding features based on multisense-scaled attention architecture to improve the predictive performance of anticancer peptides. *Bioinformatics* **2021**, *37*, 4684–4693. [CrossRef]
20. Cui, F.; Li, S.; Zhang, Z.; Sui, M.; Cao, C.; El-Latif Hesham, A.; Zou, Q. DeepMC-iNABP: Deep learning for multiclass identification and classification of nucleic acid-binding proteins. *Comput. Struct. Biotechnol. J.* **2022**, *20*, 2020–2028. [CrossRef]
21. Zheng, J.; Zhang, X.; Zhao, X.; Tong, X.; Hong, X.; Xie, J.; Liu, S. Deep-RBPPred: Predicting RNA binding proteins in the proteome scale based on deep learning. *Sci. Rep.* **2018**, *8*, 15264. [CrossRef]
22. Du, X.; Diao, Y.; Yao, Y.; Zhu, H.; Yan, Y.; Zhang, Y. DeepMVF-RBP: Deep Multi-view Fusion Representation Learning for RNA-binding Proteins Prediction. In Proceedings of the 2018 IEEE International Conference on Bioinformatics and Biomedicine (BIBM), Madrid, Spain, 3–6 December 2018; pp. 65–68.
23. Zhao, Y.; Du, X. econvRBP: Improved ensemble convolutional neural networks for RNA binding protein prediction directly from sequence. *Methods* **2020**, *181–182*, 15–23. [CrossRef] [PubMed]
24. Zhang, J.; Chen, Q.; Liu, B. iDRBP_MMC: Identifying DNA-Binding Proteins and RNA-Binding Proteins Based on Multi-Label Learning Model and Motif-Based Convolutional Neural Network. *J. Mol. Biol.* **2020**, *432*, 5860–5875. [CrossRef] [PubMed]
25. Pan, X.; Fan, Y.X.; Jia, J.; Shen, H.B. Identifying RNA-binding proteins using multi-label deep learning. *Sci. China Inf. Sci.* **2019**, *62*, 19103. [CrossRef]
26. Niu, M.; Wu, J.; Zou, Q.; Liu, Z.; Xu, L. rBPDL: Predicting RNA-Binding Proteins Using Deep Learning. *IEEE J. Biomed. Health Inform.* **2021**, *25*, 3668–3676. [CrossRef]
27. Chawla, N.V.; Bowyer, K.W.; Hall, L.O.; Kegelmeyer, W.P. SMOTE: Synthetic Minority Over-sampling Technique. *J. Artif. Intell. Res.* **2002**, *16*, 321–357. [CrossRef]