Interdisciplinary Sciences: Computational Life Sciences (2023) 15:189–201
https://doi.org/10.1007/s12539-023-00552-4

ORIGINAL RESEARCH ARTICLE

sORFPred: A Method Based on Comprehensive Features and Ensemble Learning to Predict the sORFs in Plant LncRNAs

Ziwei Chen[1,2] · Jun Meng[1,2] · Siyuan Zhao[1,2] · Chao Yin[1,2] · Yushi Luan[1,2]

Received: 22 July 2022 / Revised: 11 January 2023 / Accepted: 13 January 2023 / Published online: 27 January 2023

Abstract

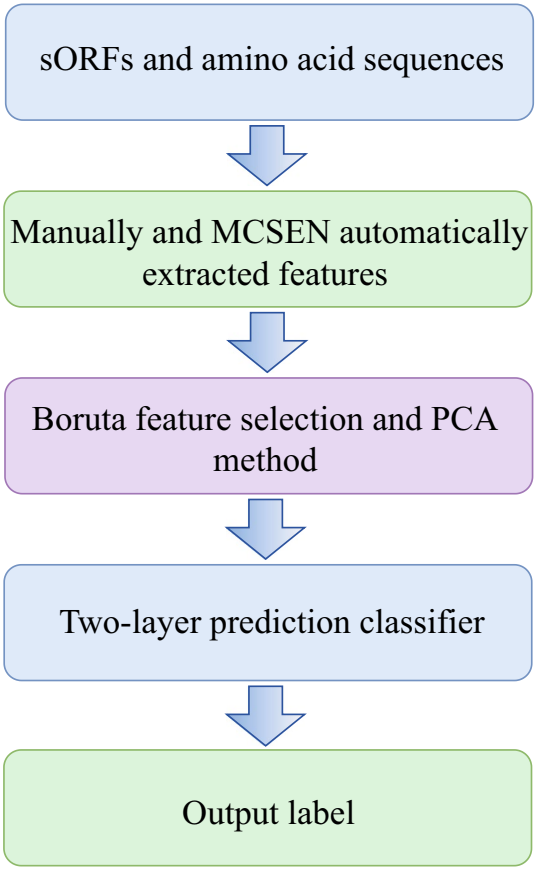
Long non-coding RNAs (lncRNAs) are important regulators of biological processes. It has recently been shown that some lncRNAs include small open reading frames (sORFs) that can encode small peptides of no more than 100 amino acids. However, existing methods are commonly applied to human and animal datasets and still suffer from low feature representation capability. Thus, accurate and credible prediction of sORFs with coding ability in plant lncRNAs is imperative. This paper proposes a new method termed sORFPred, in which we design a model named MCSEN by combining multi-scale convolution and Squeeze-and-Excitation Networks to fully mine distinct information embedded in sORFs, integrate and optimize multiple sequence-based and physicochemical feature descriptors, and built a two-layer prediction classifier based on Bayesian optimization algorithm and Extra Trees. sORFPred has been evaluated on sORFs datasets of three species and experimentally validated sORFs dataset. Results indicate that sORFPred outperforms existing methods and achieves 97.28% accuracy, 97.06% precision, 97.52% recall, and 97.29% F1-score on *Arabidopsis thaliana*, which shows a significant improvement in prediction performance compared to various conventional shallow machine learning and deep learning models.

✉ Jun Meng
mengjun@dlut.edu.cn

[1] School of Computer Science and Technology, Dalian University of Technology, Dalian 116024, Liaoning, China

[2] School of Bioengineering, Dalian University of Technology, Dalian 116024, Liaoning, China

Graphical Abstract



Keywords Comprehensive features · Ensemble learning · sORFs · Small peptides · LncRNAs

1 Introduction

Long non-coding RNAs (lncRNAs) with biological functions in animals [1–3] and plants [4–6] have been discovered in recent years. In addition, it has been found that lncRNAs play a wide range of roles in many processes of individual development, such as transcription and inactivation of chromosomes, gene expression and shutdown, and cell cycle [7]. Increasingly, it has been shown that a number of lncRNAs with small open reading frames (sORFs) of no more than 300 nucleotides (nt) in length can encode small peptides of no longer than 100 amino acids (aa) [8, 9]. The first small peptide encoded by the sORF in plants lncRNAs was found to be the soybean ENOD40 peptide, which regulates the conversion and uptake of sucrose in root nodules in legumes [10]. Frank et al. identified a small new protein which can promote division and polarized growth of maize leaf epidermal cells [11]. Li et al. found that small peptides encoded by sORFs in plant lncRNAs can regulate plant organogenesis and leaf morphogenesis [12]. In Drosophila, a lncRNA was found to encode three 11aa and one 32aa small peptides that function during the epidermal morphogenesis of embryonic development by regulating the structure of the F-actin bundle [13]. According to Pauli and colleagues' findings, during zebrafish gastrulation, a small peptide known as Toddler which is encoded by the sORF of lncRNA, can stimulate cell motility by activating APJ/Apelin receptor signaling [14]. Olson et al. identified a small peptide DWORF with a length of 34aa encoded by the sORF located on lncRNA. DWORF was shown to be abundantly expressed in the mouse heart and is able to regulate muscle contraction, and its expression was found

to be suppressed in human ischemic heart tissue, suggesting that it may be involved in heart failure [3]. Matsumoto et al. found that lncRNA LINC00961 has the ability to encode a small peptide called SPAR, which inhibits the activity of mTORC1 and thus regulates muscle regeneration [15].

As more and more small peptides are discovered, related research has attracted more and more attention. The current research on sORFs is mainly carried out through computational prediction and biological experiments. Biological experiments mainly include ribosome profiling [16–18], mass spectrometry [19, 20], and immunoblot assays [21]. Due to the short length, small relative mass, and large order of magnitude of sORFs and small peptides, biological experiments have many limitations, such as time-consuming, inefficient, inaccurate, costly, and difficult to achieve batch identification. With the speedy development of machine learning algorithms, it has played an important role in lncRNA-disease associations [22], cell-penetrating peptides identification [23], lncRNA identification [24], miRNA-lncRNA interaction [25], and DNA–protein binding sites prediction [26]. Moreover, it can provide a powerful reference for biological experimental validation, saving a lot of time and cost, and accelerating the pace of research.

Machine learning-based small peptides identification is still in its inception stage. CRITICA [27], CPC2 [28], and PhyloCSF [29], which are alignment-based methods used to distinguish mRNAs from lncRNAs, can be used to identify small peptides. The fact that these alignment-based methods heavily rely on pre-existing data is, however, a clear disadvantage of these approaches. If there is a significant gap between the fresh data and the historical data, the outcomes of the prediction will suffer as a direct consequence. The other is the alignment-free method, which only depends on the intrinsic information of the sequence, making them more flexible and general than the alignment-based methods. MiPepid, a tool designed exclusively for recognizing micropeptides, was created by Zhu et al. [30], using 4-mers features to construct logistic regression (LR) models. It has better performance compared to tools such as CPC, CPC2, CPAT, not only in predicting regular-sized proteins, but also in identifying micropeptides well. Tong et al. [31] proposed a feature engineering CPPred using 8 RNA sequence-based features and protein sequence-based features collected from CPAT and CPC2 with the addition of CTD features to identify coding RNA using SVM. In addition, it does a good job of distinguishing between coding and non-coding RNAs of lengths less than 303nt. Zhang et al. [32] adopted the dataset of CPPred in their study, extracting and integrating multiple sequence basic composition features as well as the newly proposed nucleotide bias descriptor. The mDS feature selection method was then used to filter the features before they were fed into a CNN, and thus the CNN-based RNA coding potential prediction method DeepCPP was proposed. Notably, DeepCPP overcomes the sORF mRNA identification barrier by not only performing well on normal data but also on sORF-type data in particular. In addition, the authors collected 8 small peptides encoded by ncRNAs that are associated with cancers or diseases for the experiments. This further demonstrates the good performance of DeepCPP.

The above methods have been of great help in my research since they have produced outstanding results in identifying small peptides and discriminating between coding and non-coding RNAs. However, it is noteworthy that the existing methods use a low number of features, and the lack of research on feature representation capability hinders further improvement of prediction performance. If the discriminative information can be fully mined from multiple perspectives, the prediction performance is expected to be further improved. Second, existing methods are generally single shallow machine learning models or deep learning models, such as XGBoost, SVM, and CNN. Single classifiers have their own drawbacks, and this is where further improvements can be made.

For reasons such as data sample size, the current focus of small peptides research has been skewed toward humans and animals, while relatively little research has been done on plants. Since there are differences in the way that ncRNAs are produced in plants and animals [33], there may likewise be some differences between the small peptides encoded by sORFs in plant and animal lncRNAs. Thus, whether such predictors trained using human or animal datasets can be used directly for studies related to small peptides encoded by sORFs in plant lncRNAs is a question that needs to be validated. Therefore, it is imperative to develop a method suitable for predicting sORFs in plant lncRNAs. Accurate and effective prediction of sORFs with coding potential in plant lncRNAs will not only lay the foundation for further identifying small peptides with biological functions encoded by sORFs in lncRNAs, but also be of great importance for studies such as plant breeding and exploring plant biological processes.

To address the aforementioned challenges, a brand-new ensemble learning-based method termed sORFPred is proposed. The following aspects sum up the uniqueness of sORFPred. (1) A model based on multi-scale convolution and Squeeze-and-Excitation Networks (SENet) [34] named MCSEN is designed to extract generative high-level features. (2) To fully mine the discriminative information of sORFs from various perspectives, a multi-feature integration strategy is used to fuse 16 sequence-based and physicochemical descriptors with generative high-level features to obtain 2307 dimensional features. (3) Principal component analysis (PCA) is utilized to optimize the feature space and a novel feature selection method Boruta [35] is adopted to remove redundant features. (4) To obtain

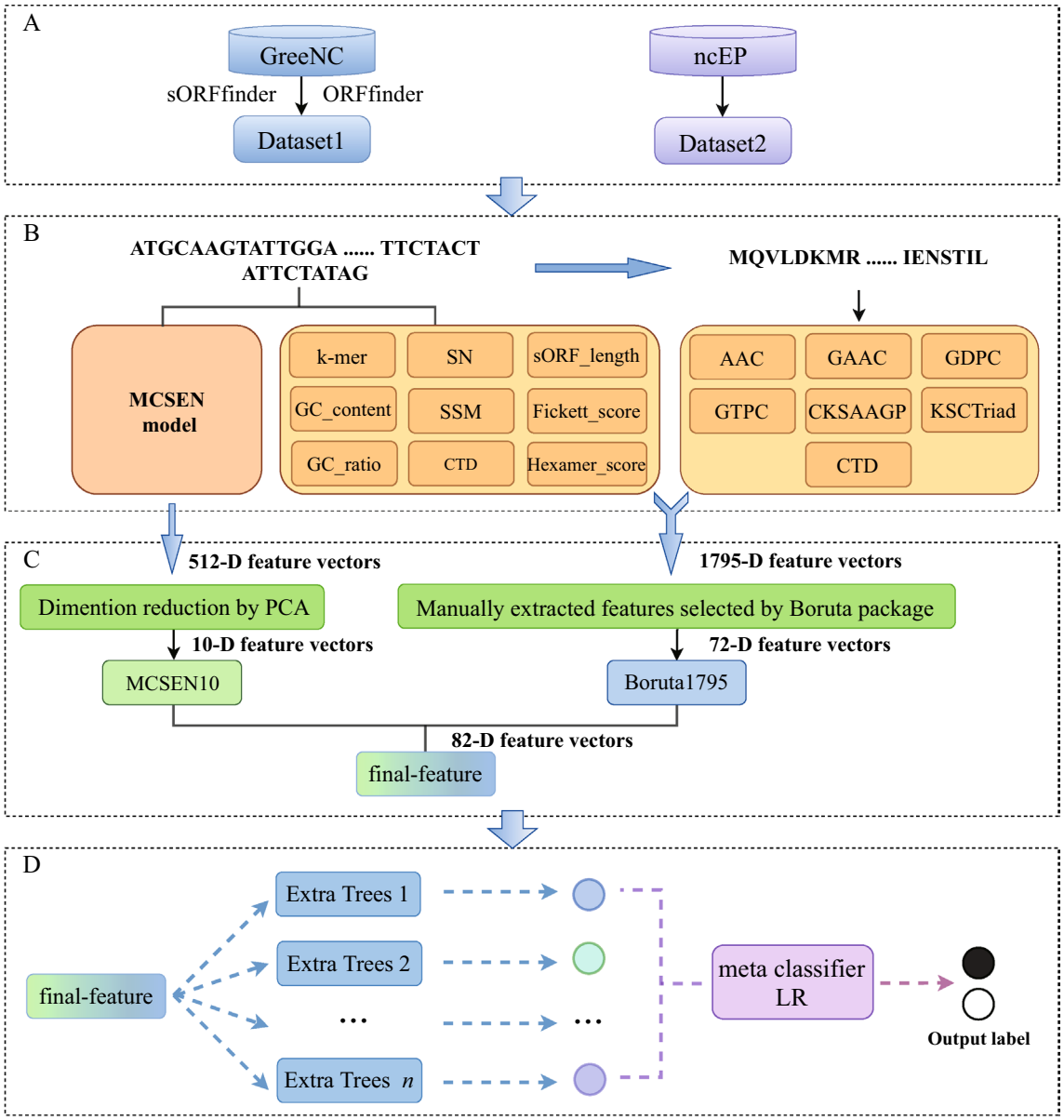


Fig. 1 The overall architecture of sORFPred method. It comprises four phases: **A** dataset construction. **B** Feature extraction. **C** Feature optimization. **D** Building ensemble model

accurate and robust results, base classifiers are optimized by the Bayesian optimization package [36], and then are combined by LR to form the final predictor. sORFPred has verified its performance and generalization capabilities by comparing with several existing methods. The results show that sORFPred outperforms shallow machine learning as well as deep learning models, with an accuracy of 97.28% on the *Arabidopsis thaliana* (*A. thaliana*) dataset.

The rest of this paper is structured as follows. Section 2 provides an overview of the dataset acquisition, feature engineering, and sORFPred's framework. Subsequently, Sect. 3 analyzes and discusses the experimental results. Lastly, the presented work is summarized in Sect. 4 along with a preliminary discussion of future work.

2 Materials and Methods

2.1 Framework of sORFPred

An ensemble learning method called sORFPred is proposed in this paper, and its framework is shown in Fig. 1. Method sORFPred consists of three phases: (1) Feature

extraction, (2) Feature optimization, and (3) Ensemble classification. In phase (1), sORF sequence is encoded into a 168-dimensional feature vector by 9 nucleotide sequence-based descriptors, while the amino acid sequence corresponding to each sORF sequence is encoded into a 1627-dimensional feature vector by 7 amino acid sequence-based descriptors, totally 1795-dimensional feature vector by manually extracting feature descriptors. Further, the MCSEN model is used to convert each sORF sequence into a 512-dimensional feature vector. In phase (2), the Boruta package and the PCA method will work together to optimize the original features. After this, the classifiers in phase (3) will make their predictions based on the final feature vectors. The predictor is a two-layer prediction model using the stacking strategy. The first layer uses Extra Trees as base classifiers which are optimized by the Bayesian optimization package, then the second layer uses LR to combine the base classifiers.

2.2 Datasets Construction

Currently, due to the lack of sORFs that have been experimentally validated, the credible datasets are constructed with the help of available bioinformatics tools and public databases. *A. thaliana* as the most widely used model plant has been intensively studied. *Glycine max (G. max)* and *Physcomitrella patens* (*P. patens*) also have relatively abundant data, which have been commonly used in previous studies [37]. Therefore, the lncRNA data of those species were downloaded from GreeNC [38]. Then, sORFfinder [39] and ORF finder [40] were then used to obtain sORFs. After obtaining the intersection and difference sets of the results from the two tools, the sequences with similarities higher than 80% were removed using CD-HIT [41]. Since sORFfinder can predict sORFs with coding ability, the intersection is used as the candidate positive sample set while the difference set is taken as the candidate negative sample set. Then, based on the idea of logical reasoning [42], the knowledge base was built to further filter the candidate positive sample set and candidate negative sample set to improve the credibility of the dataset, and thus Dataset1 was obtained. In addition, Dataset2 was constructed to test the performance and generalization ability of sORFPred. 20 sORFs sequences of functional lncRNA-encoded small peptides from *Drosophila melanogaster (D. melanogaster), Homo sapiens (H. sapiens)*, *Mus musculus (M. musculus)*, *G. max*, *Zea mays (Z. mays)*, and *A. thaliana* were downloaded from ncEP [43] as the positive samples while 40 sORFs without coding potential that do not belong to Dataset 1 are picked at random as the negative sample set. A summary of the details of all datasets is presented in Table 1.

Table 1 Datasets information

Dataset	Plant species	Positive sample	Negative sample
Dataset1	*A. thaliana*	2300	2300
	P. patens	6000	6000
	G. max	3500	3500
Dataset2	Validated sORFs	20	40

Table 2 Features information

Feature category	Feature name	Dimension
Nucleotide sequence-based features	1-mer	4
	2-mer	16
	3-mer	64
	GC_content	1
	GC_ratio	1
	SN	1
	SSM	48
	sORF_length	1
	Fickett score	1
	Hexamer score	1
	CTD	30
Amino acid sequence-based features	AAC	20
	GAAC	5
	GDPC	25
	GTPC	125
	CTD	273
	CKSAAGP ($k=0$)	25
	CKSAAGP ($k=1$)	25
	CKSAAGP ($k=2$)	25
	CKSAAGP ($k=3$)	25
	CKSAAGP ($k=4$)	25
	CKSAAGP ($k=5$)	25
	KSCTriad ($k=0$)	343
	KSCTriad ($k=1$)	343
	KSCTriad ($k=2$)	343
Features extracted by MCSEN	MCSEN512	512

2.3 Feature Extraction

A multi-feature integration strategy is used to fuse various feature descriptors to fully mine the discriminative information of sORFs from different perspectives. Based on sequence categories and extraction methods, these feature descriptors can be divided into the following three major categories: nucleotide sequence-based features, amino acid sequence-based features, and features extracted by the MCSEN model. Then, the sORF sequences and corresponding amino acid sequences were successfully encoded with the 1795-dimensional manually extracted

feature vectors and 512-dimensional MCSEN-extracted feature vectors. All features are summarized in Table 2. The details of how these feature descriptors are encoded can be found in the Supplementary Method S1. However, the performance of the model will be constrained by the high feature dimensionality and the superfluous features. Therefore, a feature selection strategy is described in the feature optimization part in order to optimize the feature space.

2.3.1 Nucleotide Sequence-Based Features

In order to predict the sORFs with coding potential, the sequence-based features of sORFs were extracted based on the traditional feature extraction method of RNA, including k-mer [44], short sequence motifs (SSM) [45], signal-to-noise (SN) [46], the content of base C and G (GC_content), the ratio of base C and G (GC_ratio), and the length of the sequence (sORF_length). In addition, we extracted some RNA features descriptors from recently published work [31, 47] and used them to predict sORF for the first time. Fickett score and Hexamer score are derived from CPAT [47]. Similarly, the CTD descriptor mentioned by CPPred [31] is also added.

A total of 168-dimensional features has been extracted for sORF sequences, where k-mer, as an approximate expression of codon frequencies, describes the nucleotide sequence composition information. GC_ratio and GC_content are also extracted as the genome of an organism or a specific DNA or RNA segment has a specific content of base C and G. SN descriptor can be interpreted as strength of the 3-base periodicity per nucleotide and indicates the bias of base usage in sORFs. Since k-mer descriptor only considers the properties of contiguous nucleotides, SSM descriptor is introduced to describe the association between discontinuous nucleotides. The difference in the combined effect of nucleotide composition and codon use bias in sORF sequences is described by the Fickett score descriptor. It is calculated from the sORF sequences using four position values together with four composition values. Hexamer score descriptor distinguishes coding sequences from non-coding sequences based on hexamer usage bias, while the hexamer usage difference between coding and non-coding sequences is measured by the log-likelihood ratio. In addition, the CTD descriptor describes the differences in nucleotide composition, nucleotide transition, and nucleotide distribution between coding and non-coding sequences.

2.3.2 Amino Acid Sequence-Based Features

First, the sORFs sequences in the dataset are translated into amino acid sequences based on the correspondence between codons and amino acids. As for amino acid sequences, seven descriptors have been collected from iFeature [48]. 1627-dimensional features are extracted for the amino acid sequences, where Amino Acid Composition (AAC) describes the composition frequencies of 20 amino acids. Based on the dipoles and side chain volumes, the 20 amino acids can be categorized into 7 groups. The *k*-Spaced Conjoint Triad (KSCTriad) descriptor treats any three amino acids separated by k ($k=0, 1, 2$) residues as a single unit when considering the properties of an amino acid and its neighbors. The Composition, Transition, and Distribution (CTD) descriptor categorizes the 20 amino acids into 3 groups based on 13 physicochemical attributes, which indicate the amino acid distribution patterns of a certain structural or physicochemical feature in a peptide or protein sequence. On the basis of physical properties, such as hydrophobicity, charge, and molecular size, the 20 amino acids are further classified into 5 categories. Then, the frequency of each amino acid group is represented by the Grouped Amino Acid Composition (GAAC) descriptor. In addition, Grouped Dipeptide Composition (GDPC) and Grouped Tripeptide Composition (GTPC) descriptors are used to define grouped dipeptide composition and grouped tripeptide composition in an amino acid sequence, respectively. Moreover, the Composition of *k*-spaced Amino Acid Pairs (CKSAAGP) descriptor was employed to calculate the

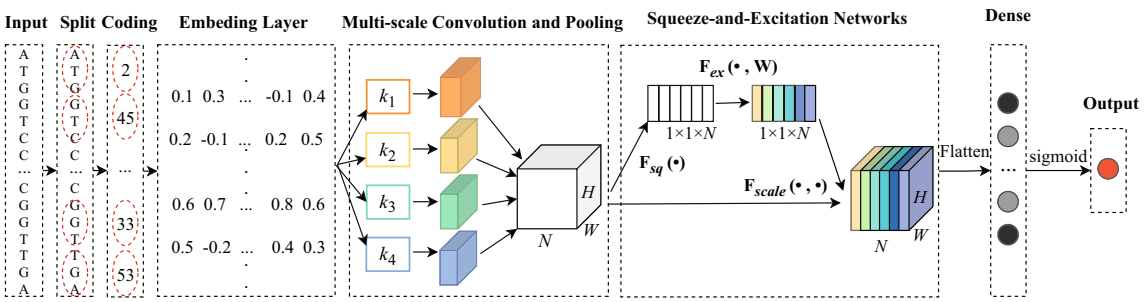


Fig. 2 Overall architecture of MCSEN model. There are three main operations: (1) encode sORFs sequence by p-nts encoding method, (2) extract local features by multi-scale convolution layer and max-pooling layer, and (3) Recalibrate channel-wise feature responses by SENet structure

frequency of amino acid group pairs separated by any k residues ($k=0, 1, 2, \ldots, 5$).

2.3.3 Features Extracted by the MCSEN Model

The features in deep learning methods are automatically extracted by the artificial neural network, which reduces human intervention and provides more feature information compared to the manually extracted features. To further obtain more information of sORF sequences, the MCSEN model combining multi-scale convolution and SENet [34] is constructed to extract 512-dimensional local features.

The traditional encoding methods tend to ignore the correlation between nucleotides. To address this problem, the p-nts [49] encoding method is used to encode sORF sequences. Instead of a single convolution kernel, a multi-scale convolution operation is used to extract features. For the purpose of solving the problem of loss resulting from the different importance of different channels during the convolution and pooling process, the SENet structure is introduced. SENet adopts a new feature rescaling strategy, which automatically obtains the importance of each channel through learning, and then enhances the useful features and suppresses the features which are useless for the problem at hand in accordance with the importance, thus highlighting the key features and further optimizing the model performance. During the training phase, different hidden neurons are randomly dropped by Dropout and the training time will be early stopped to avoid overfitting. The overall architecture is shown in Fig. 2. In addition, the feature extraction with MCSEN includes the following steps.

Step 1: The sORF sequences are split and encoded using p-nts ($p=3$) encoding method.

Step 2: The embedding layer maps the coded sequence into a 128×101 matrix vector to facilitate convolutional operations and feature extraction.

Step 3: To avoid loss of effective information, convolution kernels of 4 different scales are used to more fully extract local features. The convolution pooling operation for each scale is performed as follows.

(a) 64 convolution kernels of scale f ($K \in \Re^{m\times f}$) are selected for the convolution operation to obtain the convolved feature matrix $\boldsymbol{C}$, where m denotes the convolution kernel width which is equal to the embedding dimension and f is the convolution kernel length.

(b) Max-pooling operation is performed on the feature matrix $\boldsymbol{C}$ to extract the important feature information P in the local region, where c_i is the i-th convolution feature map, f denotes the convolution kernel scale and l is the length of the sequence. After the convolution operation with a convolution kernel of scale f, the output after the max-pooling operation with pooling size $1\times(l\text{-}f)$ is as follows:

$$P_i^{l-f} = \max\left(c_i, c_{(i+1)}, \ldots, c_{(i+l-f-1)}\right), \quad i \in (1, 2, \ldots, f+1) \tag{1}$$

(c) After performing the convolution and pooling operation on the 4 scales of convolution kernels f_1, f_2, f_3, and f_4, the output results of each are concatenated to obtain the final result V of the multi-scale convolution operation, which is represented as follows:

$$V = \left[P^{l-f_1}, P^{l-f_2}, P^{l-f_3}, P^{l-f_4}\right] \tag{2}$$

Step 4: Input V into the SENet structure to recalibrate channel-wise feature responses.

First, the feature map with input size $W\times H\times N$ is squeezed, that is, the global average pooling is performed (pooling size is $h\times w$), and then the feature map is compressed to $1\times1\times N$ vectors. Subsequently, a two-layer fully connected bottleneck structure is used for the excitation operation to determine the weights of each channel in the feature map. In addition, the number of channels is reduced by the SERatio parameter to reduce the computation. SERatio is set to 1/58 in this paper. Finally, the result is output after the weight value for each channel determined by the SENet structure has been multiplied by the 2-D matrix of the corresponding channel in the original feature map.

Step 5: The results obtained in step 4 are input to the Flatten layer, which turns the multidimensional input into one dimension.

Step 6: Then, the Dense layer with the parameter 1 is connected, and the feature vector is mapped to [0, 1] to get the probability of the predicted label after the activation function sigmoid.

Step 7: Finally, the sORF sequences are fed into the MCSEN model to extract local features, and the output of the Flatten layer is then extracted to obtain 512-dimensional features.

2.4 Feature Optimization

Boruta package [35] differs from the common feature selection method. It aims to pick all features which are associated with the dependent variable. Boruta package is based on the idea of shadow features and binomial distribution and determines the importance of features by creating synthetic features consisting of the target features and their randomly rearranged values. The process of Boruta is shown in Fig. 3. In addition, the specific steps are: (1) shuffle the original feature matrix to obtain shadow features, and then a new feature matrix is formed by stitching the original features with the shadow features. (2) The newly obtained feature matrix is adopted as the input to train the classifier. (3) Calculate the importance values of the original features and the

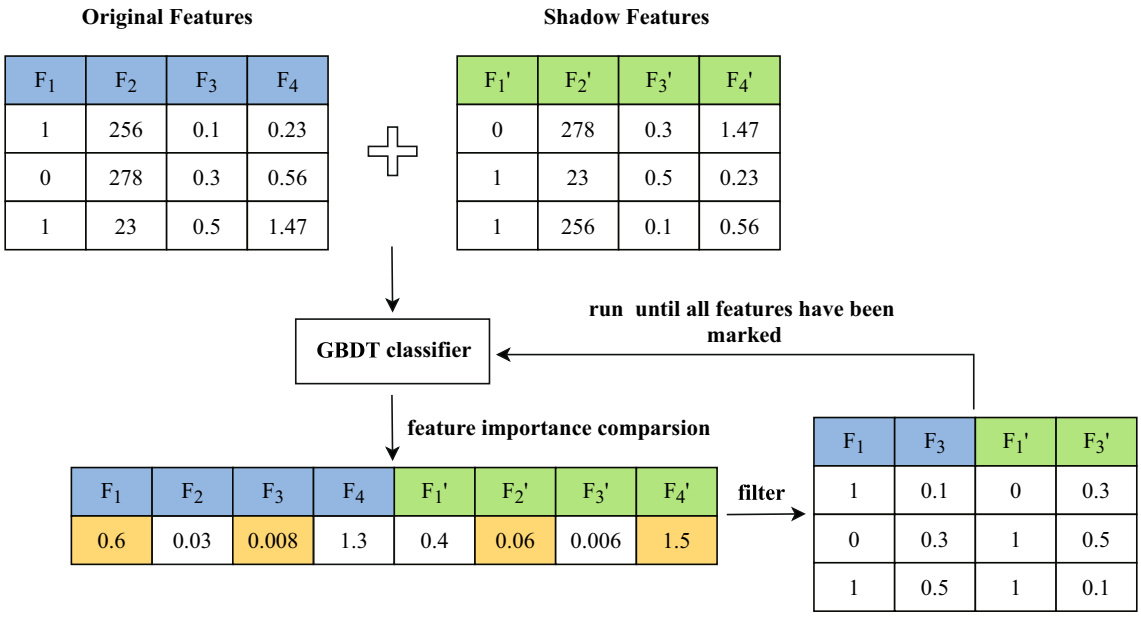


Fig. 3 Framework of Boruta feature selection method. The specific steps are: (1) shuffle the original feature matrix to obtain shadow features, (2) calculate the corresponding importance values of shadow features and original features separately, and (3) filter features based on feature importance in an iterative process

shadow features separately. (4) If the importance value of the original features is higher than the shadow features, then the feature is marked as "important" and retained. Otherwise, it is marked as "unimportant" and will be removed from the feature set. (5) Delete all shadow features. (6) Repeat the above steps until all features are marked as "important" or "unimportant".

In order to investigate the effect of each feature importance ranking algorithms on the classification results, RF, XGBoost, LightGBM and GBDT were used as feature importance ranking algorithms for feature selection under the Boruta framework, respectively. For the sake of fairness in comparison, 'n_estimators' was selected to be set to 'auto' in the Boruta framework, and 'max_depth' was set to 5 uniformly, and the filtered features were then fed into the ensemble classification model. The experimental results have been added to the Supplementary Table S1. According to the result of the experiments, it can be seen that using XGBoost and LightGBM as feature importance ranking algorithms, the number of features obtained from the filtering is small and the information contained in the features is too one-sided. Although the accuracy is improved, the generalization is relatively poor as seen from the two independent test sets. The Boruta framework using GBDT as the feature importance ranking algorithm can further remove the redundant features while retaining relatively comprehensive feature information compared to the RF feature importance ranking algorithm. Therefore, GBDT is adopted as the feature importance ranking algorithm under the Boruta framework.

To enhance the model performance and better understand the features of the data, the manually extracted 1795-dimensional features are filtered by the Boruta package to obtain all features useful for prediction (Boruta1795). To remove redundant data as well as prevent the overfitting phenomenon, the features extracted by MCSEN (MCSEN512) are dimension-reduced by PCA to obtain a new feature set (MCSEN10). Then, Boruta1795 is combined with MCSEN10 to form the final feature set.

2.5 Bayesian Optimization Method

Bayesian optimization method [36] builds probabilistic models based on the information available from previous evaluations of the objective function and finds the value of minimizing or maximizing the objective function by a minimum number of steps. Compared with the currently used algorithms such as particle swarm optimization, random search, genetic algorithm, and grid search, the Bayesian optimization method considers the previous parameter information and constantly updates the prior, which has fewer iterations and better performance, and can save a lot of useless work.

2.6 Ensemble Learning Construction

Ensemble models can make stronger and more accurate predictions compared with single classifier, because each classifier in the ensemble model has its own strengths. Ensemble models have many successful applications in bioinformatics [50–52], and the stacking method is a common integration strategy. A two-layer stacking strategy is used in this paper. First, the final features are input into several shallow machine learning models, and the performance of each model is measured by the five-fold cross-validation method. Then, the model which has the best prediction performance is selected and further optimized by the Bayesian optimization method to obtain the first layer's basic classifiers. Then, the prediction results of each base classifier are input into the LR model of the second layer to obtain the final prediction results.

2.7 Implementation of sORFPred

MCSEN is implemented by Keras 2.2.4 with the backend of TensorFlow 1.12.2. The scripts are written by Python 3.6.5. While sORFPred is implemented by Keras 2.7.0 with the backend of TensorFlow 2.7.0. The scripts are written by Python 3.8.5. The hardware experiment environment is a PC equipped with 16 GB of RAM, the GPU is AMD Radeon R7 200 series, and the CPU is 4 cores of Intel Core i5-6500 3.2 GHz.

2.8 Evaluation Criteria

In this paper, four commonly used evaluation criteria are used to evaluate the performance of sORFPred. They are formulated as follows:

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \tag{3}$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{4}$$

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{5}$$

$$\text{F1 - score} = \frac{2\text{TP}}{2\text{TP} + \text{FP} + \text{FN}} \tag{6}$$

where TP, TN, FP, and FN stand for the corresponding totals of true-positive, true-negative, false-positive, and false-negative samples, respectively. As for all the metrics listed above, the better the model performs, the higher the score it receives.

3 Results

3.1 Performance Analysis of the Model in Different Feature Spaces

This section analyzes the performance of various types of features on the *A. thaliana* dataset. In the feature representation stage, three categories of features were extracted for encoding sORF sequences, namely nucleotide sequence-based features (nt168), amino acid sequence-based features (aa1627), and features extracted by the MCSEN model (MCSEN512), respectively. Further, the dimension of MCSEN512 is reduced using PCA to obtain a new feature set (MCSEN10). The nucleotide sequence-based and amino acid sequence-based features are fused to obtain extracted features of 1795 dimensions (original1795).

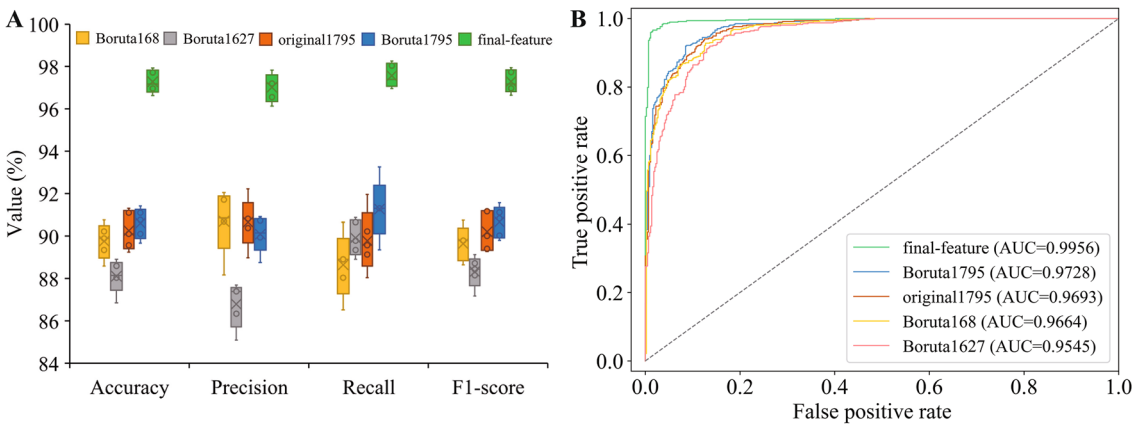


Fig. 4 Results of the proposed sORFPred with different types of features on *A. thaliana* dataset. **A** The performances of sORFPred with different types of features. **B** ROC curves and AUC values of sORFPred with different types of features

Then, original1795 is filtered using Boruta package and fused with MCSEN10 to obtain the final feature set (final-feature). In this section, it will be discussed which type of feature is more discriminative in predicting sORFs with coding potential and sORFs without coding ability. For a fair comparison, the nt168, aa1627, and the original1795 feature are processed separately by the Boruta package to obtain the optimized feature subsets Boruta168, Boruta1627, and Boruta1795. In addition, five-fold cross-validation is conducted to compare the performance of ensemble model with different types of features. The performance is shown in Fig. 4A. In addition, receiver-operating characteristic (ROC) curves is further plotted as shown in Fig. 4B in order to present the comparison results more clearly. Detailed results are shown in Supplementary Table S2. It is clear that all five types of features are effective in predicting sORFs. Notably, the fused feature Boruta1795 achieved better results than the single feature Boruta168 and Boruta1627, while the final-feature after fusing MCSEN10 achieved an average accuracy of 97.28% for the prediction of sORFs, which was higher than the manually extracted features (Boruta168, Boruta1627, and Boruta1795) of 6.67~9.17%. This suggests that the MCSEN model can successfully learn the local features of sORF sequences. It can also be seen that the features optimized by Boruta package (Boruta1795) achieved better performance compared with the original features (original795), while reducing the feature dimensionality and running time.

3.2 Selection of Base Classifiers

In order to obtain the optimal base classifiers, GaussianNB, kNN [53] SVM [54], RF [55], and Extra Trees [56] are selected as candidate classifiers. Then, the performance of each classifier is evaluated using the five-fold cross-validation method on *A. thaliana* dataset. As shown in Supplementary Table S3, Extra Trees achieved the best performance compared to other models with 96.09% Accuracy, 95.31% Precision, 97.26% Recall, and 96.32% F1-score. In terms of Accuracy, Extra Trees is higher than the other models by 0.24–9.57% and the standard deviation (SD) is only 0.51%, indicating that the stability of the Extra Trees is better. Overall, RF, Extra Trees, and kNN outperformed the other models by a large margin. Although RF is slightly higher than Extra trees in terms of precision, Extra Trees outperforms RF in terms of Accuracy, Recall, and F1-score by 0.31%, 2.74%, and 1.4%, respectively. Therefore, Extra Trees with relatively better performance is identified as the base classifiers.

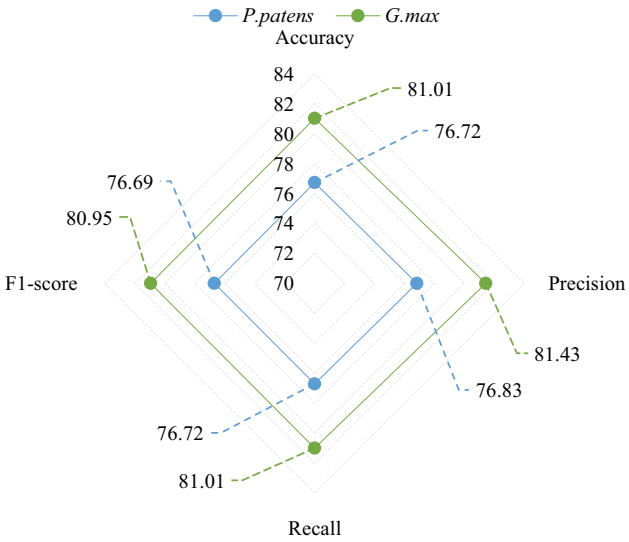


Fig. 6 Performance of sORFPred on other species

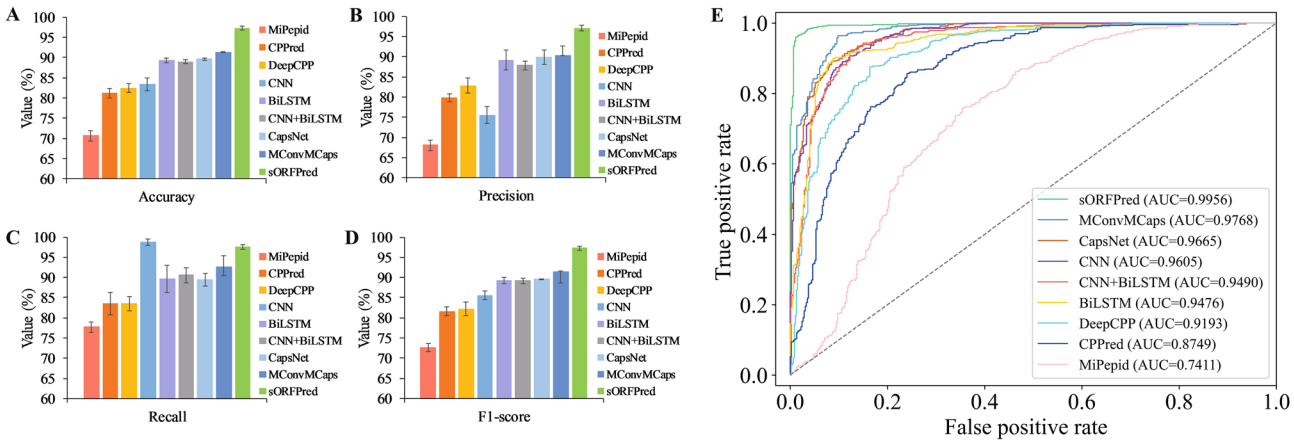


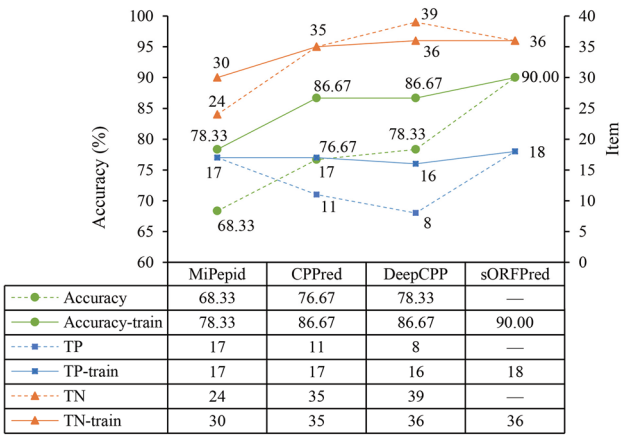
Fig. 5 Performance of the proposed sORFPred and other models on *A. thaliana* dataset in terms of **A** accuracy, **B** precision, **C** recall, **D** F1-score and **E** ROC curve

3.3 Comparison with Other Models

To more impartially verify the effectiveness of sORFPred, on the *A. thaliana* dataset, it was compared with commonly used deep learning models, namely CNN, BiLSTM, CNN + BiLSTM, CapsNet, and MConvMCaps [42], as well as the state-of-the-art methods, namely MiPepid, CPPred, and DeepCPP. The performance of each model is presented in Fig. 5 and more specific data are available in Supplementary Table S4. As can be seen in Fig. 5, sORFPred performs obviously better than those models. The mean of Accuracy, Precision, and F1-score are 97.28%, 97.06%, and 97.29%, respectively, which are 5.98~26.63%, 6.71~28.94%, and 5.88~24.7% higher than the compared models. Although sORFPred is slightly lower than CNN in terms of Recall, the high recall value of CNN is obtained at the expense of the precision of prediction. Overall, sORFPred is more powerful in distinguishing whether sORFs have coding ability or not than those commonly used deep learning models and the state-of-the-art methods. From Fig. 5E, it is clear that area under the curve of sORFPred is significantly larger than the area under the other curves. This indicates that the proposed method has high sensitivity and a low false-positive rate. In other words, sORFPred can better learn the information embedded in the original data so as to achieve a robust and credible prediction of sORFs.

3.4 Prediction Performance on Other Species

In order to validate the generalization capability of sORFPred, experiments are conducted on *P. patens* and *G. max* datasets, respectively. As shown in Fig. 6, the model trained on *A. thaliana* datasets was then tested on *P. patens* and *G. max*, respectively, with accuracies of 76.72% and 81.01%, indicating that sORFPred generalizes well to other plants.



	MiPepid	CPPred	DeepCPP	sORFPred
Accuracy	68.33	76.67	78.33	—
Accuracy-train	78.33	86.67	86.67	90.00
TP	17	11	8	—
TP-train	17	17	16	18
TN	24	35	39	—
TN-train	30	35	36	36

Fig. 7 Performance of sORFPred compared to the state-of-the-art methods

3.5 Comparison with the State-of-the-Art Methods

In order to further validate the performance of sORFPred, it has been compared with commonly used methods such as MiPepid, CPPred, and DeepCPP on Dataset2, which is composed of sORFs with validated coding capabilities. Two experiments were conducted on Dataset2. One was a direct prediction of Dataset2 using the three existing tools. The other was to retrain the existing tool on the *A. thaliana* dataset before making predictions on the sORFs in Dataset2. As can be seen in Fig. 7, without retraining, although MiPepid and sORFPred correctly predicted the highest number of samples out of a total of 20 positive samples (Dataset2), MiPepid has a false-positive rate of 40%. As for the prediction of negative samples, DeepCPP predicted 39 out of a total of 40 negative samples (Dataset2). It was slightly better than sORFPred, but it was a poor predictor of positive samples with a high false-negative rate. After retraining the three tools mentioned above, although their performance improved significantly, sORFPred's performance remained relatively good. It is also clear from the comparison of the two experiments that the existing tools before being retrained do not perform well in predicting sORFs in lncRNAs due to their original datasets, and further demonstrates that sORFPred is a good method in predicting sORFs in lncRNAs.

4 Conclusions

According to our best knowledge, this research is the first to predict sORFs with coding potential in plant lncRNAs using such comprehensive and detailed features and an ensemble learning model based on the Bayesian optimization method. In comparison to existing methods, it achieves greater performance and generalization capability. We expect that sORFPred will become a potent method for the large-scale prediction of sORFs. The prediction of sORFs with coding ability in plant lncRNAs will not only lay the foundation for the discovery of lncRNA-encoded small peptides, but also provide an important reference for biological experimental validation, which is conducive to revealing the molecular mechanisms of life-form traits and disease resistance, and is of great value in agriculture and forestry production and other fields. In this research area, the majority of techniques currently used to construct predictors using a single classification algorithm, such as RF or SVM. In fact, it has been demonstrated that well-established ensemble classifiers can increase the prediction quality in protein fold classification, DNA-binding protein prediction, and other applications. Our

future research will primarily concentrate on investigating more effective feature selection techniques and more potent classification algorithms.

Supplementary Information The online version contains supplementary material available at https://doi.org/10.1007/s12539-023-00552-4.

Acknowledgements This work was supported by the National Natural Science Foundation of China (Nos. 32072592 and 32230091).

Data availability Datasets and associated source codes of sORFPred are freely available for download at https://github.com/orangewindczw/sORFPred.

Declarations

Conflict of Interest The authors declare that they have no conflicts of interest.

Ethical Approval Not applicable.

Consent to Participate Not applicable.

Consent for Publication Not applicable.

References

1. Canzio D, Nwakeze CL, Horta A et al (2019) Antisense lncRNA transcription mediates DNA demethylation to drive stochastic protocadherin α promoter choice. Cell 177:1–15. https://doi.org/10.1016/j.cell.2019.03.008
2. Hon C-C, Ramilowski JA, Harshbarger J et al (2017) An atlas of human long non-coding RNAs with accurate 5′ ends. Nature 543:199–204. https://doi.org/10.1038/nature21374
3. Nelson BR, Makarewich CA, Anderson DM et al (2016) A peptide encoded by a transcript annotated as long noncoding RNA enhances SERCA activity in muscle. Science 351:271–275. https://doi.org/10.1126/science.aad4076
4. Cui J, Luan Y, Jiang N et al (2017) Comparative transcriptome analysis between resistant and susceptible tomato allows the identification of lncRNA16397 conferring resistance to Phytophthora infestans by co-expressing glutaredoxin. Plant J 89:577–589. https://doi.org/10.1111/tpj.13408
5. Cui J, Jiang N, Meng J et al (2019) LncRNA33732-respiratory burst oxidase module associated with WRKY1 in tomato-*Phytophthora infestans* interactions. Plant J 97:933–946. https://doi.org/10.1111/tpj.14173
6. Hong Y, Zhang Y, Cui J et al (2022) The lncRNA39896–miR166b–HDZs module affects tomato resistance to *Phytophthora infestans*. J Integr Plant Biol 64:1979–1993. https://doi.org/10.1111/jipb.13339
7. Storz G (2002) An expanding universe of noncoding RNAs. Science 296:1260–1263. https://doi.org/10.1126/science.1072249
8. Röhrig H, Schmidt J, Miklashevichs E et al (2002) Soybean ENOD40 encodes two peptides that bind to sucrose synthase. Proc Natl Acad Sci 99:1915–1920. https://doi.org/10.1073/pnas.022664799
9. Narita NN, Moore S, Horiguchi G et al (2004) Overexpression of a novel small peptide ROTUNDIFOLIA4 decreases cell proliferation and alters leaf shape in *Arabidopsis thaliana*. Plant J 38:699–713. https://doi.org/10.1111/j.1365-313X.2004.02078.x
10. Campalans A, Kondorosi A, Crespi M (2004) Enod40, a short open reading frame–containing mRNA, induces cytoplasmic localization of a nuclear RNA binding protein in *Medicago truncatula*. Plant Cell 16:1047–1059. https://doi.org/10.1105/tpc.019406
11. Frank MJ, Smith LG (2002) A small, novel protein highly conserved in plants and animals promotes the polarized growth and division of maize leaf epidermal cells. Curr Biol 12:849–853. https://doi.org/10.1016/S0960-9822(02)00819-9
12. Li J, Liu C (2019) Coding or noncoding, the converging concepts of RNAs. Front Genet 10:496. https://doi.org/10.3389/fgene.2019.00496
13. Kondo T, Hashimoto Y, Kato K et al (2007) Small peptide regulators of actin-based cell morphogenesis encoded by a polycistronic mRNA. Nat Cell Biol 9:660–665. https://doi.org/10.1038/ncb1595
14. Pauli A, Norris ML, Valen E et al (2014) Toddler: an embryonic signal that promotes cell movement via Apelin receptors. Science 343:1248636. https://doi.org/10.1126/science.1248636
15. Matsumoto A, Pasut A, Matsumoto M et al (2017) mTORC1 and muscle regeneration are regulated by the LINC00961-encoded SPAR polypeptide. Nature 541:228–232. https://doi.org/10.1038/nature21034
16. Erhard F, Halenius A, Zimmermann C et al (2018) Improved Ribo-seq enables identification of cryptic translation events. Nat Methods 15:363–366. https://doi.org/10.1038/nmeth.4631
17. Ingolia NT, Brar GA, Stern-Ginossar N et al (2014) Ribosome profiling reveals pervasive translation outside of annotated protein-coding genes. Cell Rep 8:1365–1379. https://doi.org/10.1016/j.celrep.2014.07.045
18. Fritsch C, Herrmann A, Nothnagel M et al (2012) Genome-wide search for novel human uORFs and N-terminal protein extensions using ribosomal footprinting. Genome Res 22:2208–2218. https://doi.org/10.1101/gr.139568.112
19. Kersten RD, Yang Y-L, Xu Y et al (2011) A mass spectrometry–guided genome mining approach for natural product peptidogenomics. Nat Chem Biol 7:794–802. https://doi.org/10.1038/nchembio.684
20. Oyama M, Kozuka-Hata H, Suzuki Y et al (2007) Diversity of translation start sites may define increased complexity of the human short ORFeome. Mol Cell Proteomics 6:1000–1006. https://doi.org/10.1074/mcp.M600297-MCP200
21. Hemm MR, Paul BJ, Schneider TD et al (2008) Small membrane proteins found by comparative genomics and ribosome binding site models. Mol Microbiol 70:1487–1501. https://doi.org/10.1111/j.1365-2958.2008.06495.x
22. Yu G, Wang Y, Wang J et al (2020) Attributed heterogeneous network fusion via collaborative matrix tri-factorization. Inf Fusion 63:153–165. https://doi.org/10.1016/j.inffus.2020.06.012
23. Wei L, Xing P, Su R et al (2017) CPPred-RF: a sequence-based predictor for identifying cell-penetrating peptides and their uptake efficiency. J Proteome Res 16:2044–2053. https://doi.org/10.1021/acs.jproteome.7b00019
24. Meng J, Kang Q, Chang Z, Luan Y (2021) PlncRNA-HDeep: plant long noncoding RNA prediction using hybrid deep learning based on two encoding styles. BMC Bioinformatics 22:242. https://doi.org/10.1186/s12859-020-03870-2
25. Kang Q, Meng J, Cui J et al (2020) PmliPred: a method based on hybrid model and fuzzy decision for plant miRNA–lncRNA interaction prediction. Bioinformatics 36:2986–2992. https://doi.org/10.1093/bioinformatics/btaa074
26. Zhang Q, Yu W, Han K et al (2021) Multi-scale capsule network for predicting DNA-protein binding sites. IEEE/ACM Trans Comput Biol Bioinform 18:1793–1800. https://doi.org/10.1109/TCBB.2020.3025579

27. Frith MC, Forrest AR, Nourbakhsh E et al (2006) The abundance of short proteins in the mammalian proteome. PLoS Genet 2:e52. https://doi.org/10.1371/journal.pgen.0020052
28. Kang Y-J, Yang D-C, Kong L et al (2017) CPC2: a fast and accurate coding potential calculator based on sequence intrinsic features. Nucleic Acids Res 45:W12–W16. https://doi.org/10.1093/nar/gkx428
29. Lin MF, Jungreis I, Kellis M (2011) PhyloCSF: a comparative genomics method to distinguish protein coding and non-coding regions. Bioinformatics 27:i275–i282. https://doi.org/10.1093/bioinformatics/btr209
30. Zhu M, Gribskov M (2019) MiPepid: MicroPeptide identification tool using machine learning. BMC Bioinformatics 20:559. https://doi.org/10.1186/s12859-019-3033-9
31. Tong X, Liu S (2019) CPPred: coding potential prediction based on the global description of RNA sequence. Nucleic Acids Res 47:e43. https://doi.org/10.1093/nar/gkz087
32. Zhang Y, Jia C, Fullwood MJ, Kwoh CK (2021) DeepCPP: a deep neural network based on nucleotide bias information and minimum distribution similarity feature selection for RNA coding potential prediction. Brief Bioinform 22:2073–2084. https://doi.org/10.1093/bib/bbaa039
33. Zhang H, He X, Zhu JK (2013) RNA-directed DNA methylation in plants: where to start? RNA Biol 10:1593–1596. https://doi.org/10.4161/rna.26312
34. Hu J, Shen L, Sun G (2020) Squeeze-and-excitation networks. IEEE Trans Pattern Anal Mach Intell 42:2011–2023. https://doi.org/10.1109/TPAMI.2019.2913372
35. Kursa MB, Rudnicki WR (2010) Feature selection with the Boruta package. J Stat Softw 36:1–13. https://doi.org/10.18637/jss.v036.i11
36. Snoek J, Larochelle H, Adams RP (2012) Practical Bayesian optimization of machine learning algorithms. In: Advances in Neural Information Processing Systems, pp 2951–2959
37. Zhang P, Meng J, Luan Y, Liu C (2020) Plant miRNA–lncRNA interaction prediction with the ensemble of CNN and IndRNN. Interdiscip Sci Comput Life Sci 12:82–89. https://doi.org/10.1007/s12539-019-00351-w
38. Gallart AP, Pulido AH, de Lagrán IAM et al (2016) GREENC: a Wiki-based database of plant lncRNAs. Nucleic Acids Res 44:D1161–D1166. https://doi.org/10.1093/nar/gkv1215
39. Hanada K, Akiyama K, Sakurai T et al (2010) sORF finder: a program package to identify small open reading frames with high coding potential. Bioinformatics 26:399–400. https://doi.org/10.1093/bioinformatics/btp688
40. Sayers EW, Barrett T, Benson DA et al (2009) Database resources of the national center for biotechnology information. Nucleic Acids Res 37:D5–D15. https://doi.org/10.1093/nar/gkn741
41. Huang Y, Niu B, Gao Y et al (2010) CD-HIT suite: a web server for clustering and comparing biological sequences. Bioinformatics 26:680–682. https://doi.org/10.1093/bioinformatics/btq003
42. Hu H, Meng J, Zhao S et al (2022) Prediction of plant lncRNA-encoded small peptides combined with multi-scale convolutional capsule network. J Zhengzhou Univ (Natl Sci Edn) 54:12–18. https://doi.org/10.13705/j.issn.1671-6841.2021214
43. Liu H, Zhou X, Yuan M et al (2020) ncEP: a manually curated database for experimentally validated ncRNA-encoded proteins or peptides. J Mol Biol 432:3364–3368. https://doi.org/10.1016/j.jmb.2020.02.022
44. Clavijo BJ, Accinelli GG, Yanes L et al (2017) Skip-mers: increasing entropy and sensitivity to detect conserved genic regions with simple cyclic q-grams. bioRxiv. https://doi.org/10.1101/179960
45. Edwards RJ, Palopoli N (2015) Computational prediction of short linear motifs from protein sequences. Comput Pept. https://doi.org/10.1007/978-1-4939-2285-7_6
46. Yin C, Yau SS-T (2007) Prediction of protein coding regions by the 3-base periodicity analysis of a DNA sequence. J Theor Biol 247:687–694. https://doi.org/10.1016/j.jtbi.2007.03.038
47. Wang L, Park HJ, Dasari S et al (2013) CPAT: coding-potential assessment tool using an alignment-free logistic regression model. Nucleic Acids Res 41:e74. https://doi.org/10.1093/nar/gkt006
48. Chen Z, Zhao P, Li F et al (2018) iFeature: a python package and web server for features extraction and selection from protein and peptide sequences. Bioinformatics 34:2499–2502. https://doi.org/10.1093/bioinformatics/bty140
49. Meng J, Chang Z, Zhang P, et al (2019) lncRNA-LSTM: prediction of plant long non-coding RNAs using long short-term memory based on p-nts encoding. In: International Conference on Intelligent Computing. https://doi.org/10.1007/978-3-030-26766-7_32
50. Wan S, Duan Y, Zou Q (2017) HPSLPred: an ensemble multi-label classifier for human protein subcellular location prediction with imbalanced source. Proteomics 17:17–18. https://doi.org/10.1002/pmic.201700262
51. Ru X, Cao P, Li L, Zou Q (2019) Selecting essential MicroRNAs using a novel voting method. Mol Ther-Nucleic Acids 18:16–23. https://doi.org/10.1016/j.omtn.2019.07.019
52. Zhang G, Liu Z, Dai J et al (2020) ItLnc-BXE: a Bagging-xgboost-ensemble method with comprehensive sequence features for identification of plant lncRNAs. IEEE Access 8:68811–68819. https://doi.org/10.1109/ACCESS.2020.2985114
53. Zhang S, Li X, Zong M et al (2017) Learning k for KNN classification. ACM Trans Intell Syst Technol TIST 8:1–19. https://doi.org/10.1145/2990508
54. Lin W, Ji D, Lu Y (2017) Disorder recognition in clinical texts using multi-label structured SVM. BMC Bioinformatics 18:1–11. https://doi.org/10.1186/s12859-017-1476-4
55. Yao D, Zhan X, Zhan X et al (2020) A random forest based computational model for predicting novel lncRNA-disease associations. BMC Bioinformatics 21:1–18. https://doi.org/10.1186/s12859-020-3458-1
56. Peng L, Yuan R, Shen L et al (2021) LPI-EnEDT: an ensemble framework with extra tree and decision tree classifiers for imbalanced lncRNA-protein interaction data classification. BioData Min 14:1–22. https://doi.org/10.1186/s13040-021-00277-4