Bioinformatics, 34(13), 2018, 2185–2194
doi: 10.1093/bioinformatics/bty085
Advance Access Publication Date: 15 February 2018
Original Paper

OXFORD

Sequence analysis

The lncLocator: a subcellular localization predictor for long non-coding RNAs based on a stacked ensemble classifier

Zhen Cao[1,†], Xiaoyong Pan[2,†], Yang Yang[3,†], Yan Huang[4] and Hong-Bin Shen[1,*]

[1]Institute of Image Processing and Pattern Recognition, Shanghai Jiao Tong University, and Key Laboratory of System Control and Information Processing, Ministry of Education of China, Shanghai 200240, China, [2]Department of Medical Informatics, Erasmus MC, Rotterdam 3014zk, The Netherlands, [3]Department of Computer Science, Shanghai Jiao Tong University, and Key Laboratory of Shanghai Education Commission for Intelligent Interaction and Cognitive Engineering, Shanghai 200240, China and [4]State Key Laboratory of Infrared Physics, Shanghai Institute of Technical Physics, Chinese Academy of Sciences, Shanghai 200083, China

*To whom correspondence should be addressed.
[†]The authors wish it to be known that, in their opinion, the first three authors should be regarded as Joint First Authors.
Associate Editor: John Hancock

Received on September 27, 2017; revised on February 9, 2018; editorial decision on February 10, 2018; accepted on February 14, 2018

Abstract

Motivation: The long non-coding RNA (lncRNA) studies have been hot topics in the field of RNA biology. Recent studies have shown that their subcellular localizations carry important information for understanding their complex biological functions. Considering the costly and time-consuming experiments for identifying subcellular localization of lncRNAs, computational methods are urgently desired. However, to the best of our knowledge, there are no computational tools for predicting the lncRNA subcellular locations to date.

Results: In this study, we report an ensemble classifier-based predictor, lncLocator, for predicting the lncRNA subcellular localizations. To fully exploit lncRNA sequence information, we adopt both *k*-mer features and high-level abstraction features generated by unsupervised deep models, and construct four classifiers by feeding these two types of features to support vector machine (SVM) and random forest (RF), respectively. Then we use a stacked ensemble strategy to combine the four classifiers and get the final prediction results. The current lncLocator can predict five subcellular localizations of lncRNAs, including cytoplasm, nucleus, cytosol, ribosome and exosome, and yield an overall accuracy of 0.59 on the constructed benchmark dataset.

Availability and implementation: The lncLocator is available at www.csbio.sjtu.edu.cn/bioinf/lncLocator.

Contact: hbshen@sjtu.edu.cn

Supplementary information: Supplementary data are available at *Bioinformatics* online.

1 Introduction

The Non-coding RNAs (ncRNAs) have been demonstrated to be important regulators rather than junk sequences in the genome (Iyer *et al.*, 2015). There are various types of ncRNAs, including rRNAs, tRNAs, microRNAs (miRNAs), small nuclear RNAs (snRNAs), small nucleolar RNAs (snoRNAs), small interfering RNAs (siRNAs), long non-coding RNAs (lncRNAs), etc. (Mattick and Makunin, 2006). Due to the inherent complexity of molecular

mechanisms and functions, lncRNAs-related studies have largely lagged behind other types of ncRNAs. However, in recent years, lncRNAs have attracted more and more attentions in life science. According to our statistics of articles searched in PubMed, around 6000 literatures have the keyword of 'lncRNA' or 'long non-coding RNA' in the title or abstract. Particularly, over 95% of them were published in the last 5 years, suggesting much attention has been paid very recently. Their essential roles in post-transcription of gene regulation, translation of genetic information and cellular signal transduction have been revealed gradually (Batista and Chang, 2013). Moreover, they have been demonstrated to be promising biomarkers for a variety of diseases (Ayers, 2013; Li *et al.*, 2013). Therefore, understanding cellular functions of lncRNAs has become one of the central tasks in the post-genomic era. Considering the costly and time-consuming wet-lab experiments, automatic computational tools are highly desired to speed up the lncRNA-related studies, e.g. for expression analysis (Thomson *et al.*, 2004), target prediction (Brennecke *et al.*, 2005), functional pathway prediction (Vlachos *et al.*, 2012), etc.

Similar to proteins, the function of lncRNAs relies on the cellular compartments where they are located, and the localization information can provide important insights into functions (Chen, 2016). The computational prediction of subcellular localization has been a hot topic in bioinformatics for the last decade, due to the difficulties in identifying subcellular location through biological experiments (Chou and Shen, 2008). However, most existing prediction tools were designed for proteins (Pierleoni *et al.*, 2011; Shen and Chou, 2007, 2009; Wan *et al.*, 2017; Zhou *et al.*, 2017). As far as we know, there has been no computational predictor for lncRNA subcellular localization. This could be due to:

1. Traditionally, most lncRNAs are regarded as being located exclusively in nucleus, functioning as the regulators of nuclear genes (Chen and Carmichael, 2010). Their diverse subcellular fates have been discovered only very recently. Especially, Cabili *et al*. conducted a large-scale study of lncRNA subcellular locations using FISH (fluorescence in situ hybridization) technique in 2015 (Cabili *et al.*, 2015). They found that a substantial proportion of lncRNAs can be transferred into cytoplasm, and some lncRNAs are even located in both the nucleus and cytoplasm.
2. The computational prediction of lncRNA subcellular location is restricted by limited information sources. As for proteins, the well-established predictors usually utilize multiple types of features, including amino acid statistical attributes (Park and Kanehisa, 2003; Shen and Chou, 2008), signal peptide (Savojardo *et al.*, 2015), functional domain (Marchler-Bauer *et al.*, 2005), gene annotation (Chou and Cai, 2003; Zhou *et al.*, 2017), etc.

Existing sequence-based protein subcellular localization predictors can be generally grouped into two categories, homolog transfer and statistical machine learning-based approaches. The former is trying to find the annotated homology protein for the query sequence from a large database, which is straightforward but may fail when no homologous protein is found or the so-called 'twilight zone' phenomenon (Nair and Rost, 2002). Due to the relative slow annotation speed for lncRNAs and the great diversity in their sequences, finding homology annotated lncRNA sequence is difficult. Considering this, the statistical machine learning is more applicable to develop a lncRNA-orientated subcellular localization predictor at current stage. In such a protocol, three issues are of crucial importance for the predictive performance, i.e. (i) the sequence feature extraction and representation, (ii) the distribution of training dataset and (iii) the model used for learning the discriminative pattern.

How to encode raw sequence data into discriminative features is a vital issue in constructing machine learning-based model. Some residue-based statistical characteristics can be used as the features, such as the *k*-mer frequencies (Park and Kanehisa, 2003). However, according to our local tests, prediction based solely on the *k*-mer sequence features is a very challenging task. One of the potential reasons is the *k*-mer features are extracted from the observed sequence, which is affected by the mutation noise. The other reason is that when we increase *k* to cover longer potential motif pattern, the feature vector dimensions increase exponentially, which may also result in an over-fitting of the prediction model.

In contrast to the hand-designed features, deep learning models can capture high-level representation automatically, those models have achieved remarkable results in different fields, including computer vision, natural language processing, speech recognition and bioinformatics (LeCun *et al.*, 2015; Min *et al.*, 2017). For instance, deep network architectures have shed new lights on the feature extraction for protein or RNA sequences in applications of secondary structure prediction (Heffernan *et al.*, 2015; Spencer *et al.*, 2015), contact map prediction (Di Lena *et al.*, 2012), lncRNA recognition (Fan and Zhang, 2015), RNA-protein binding motifs identification (Pan and Shen, 2017), ncRNA-protein interaction sequential pattern mining (Pan *et al.*, 2016), etc.

The *k*-mer and the deep architecture abstraction features are two completely different strategies for representing the lncRNA sequence into discriminative features. The former represents observed statistical characteristics, while the latter can reflect the hidden pattern behind the sequence. They complement each other and thus we incorporate both of the two features into the model construction in this study.

Secondly, statistical supervised machine learning models' performance is heavily dependent on the training dataset since they learn the distribution rules of different classes from the data. We found that the lncRNA subcelluar location dataset shows a severely imbalanced distribution. For instance, in our benchmark dataset extracted from RNAlocate database (Zhang *et al.*, 2017), the numbers of lncRNAs located in cytoplasm, nucleus, cytosol, ribosome and exosome are 301, 152, 91, 43 and 25, respectively. The largest ratio between the majority and minority classes reaches ~12:1. In such a case, most machine learning methods will have a preference to the majority classes while perform poorly on the minority classes.

In general, both under-sampling and over-sampling techniques can alleviate the impact of data imbalance. Under-sampling is to reduce the samples from the majority class to match the minority class, while the over-sampling is to increase the samples of the minority class to match the majority class. To keep all available training samples, we applied the over-sampling approach in this paper. The unsupervised over-sampling methods include ROS (Random Over-sampling), SMOTE (Synthetic Minority Over-sampling Technique) (Chawla *et al.*, 2002), etc. The ROS method replicates randomly selected samples within the minority set, and the SMOTE method creates the same number of synthetic samples according to the existing samples of the existing minority classes. In this study, we adopted a supervised over-sampling method named SOS (Supervised Over-Sampling), which is able to take into account the sample labels to create new synthetic samples (Hu *et al.*, 2014).

In order to integrate the merits of different statistical learning models, the final prediction model of the proposed lncLocator is an ensemble predictor. The lncLocator combines four learning machines using a stacked ensemble strategy. They are random forest

with features extracted by deep neural networks (RF^A), support vector machine with features extracted by deep neural networks (SVM^A), random forest with raw *k*-mer features (RF^R) and support vector machine with raw *k*-mer features (SVM^R), respectively. Our experimental results show that the final ensemble predictor is superior to any single classifier due to the reason that fusing the classifiers' diversities is helpful to enhance the prediction performance (Shen and Chou, 2006).

2 Datasets and methods

2.1 Dataset

We extracted the subcellular localization information of lncRNAs from a comprehensive database, i.e. RNAlocate (http://www.rna-society.org/rnalocate), which covers localization information of mRNAs, miRNAs, lncRNAs, etc. The current version of RNALocate houses more than 37 700 manually curated RNA-associated subcellular localization entries with experimental evidence. It covers more than 21 800 coding and non-coding RNAs with 42 subcellular locations across 65 species, mainly including Homo sapiens and Mus musculus (Zhang *et al.*, 2017). The construction of the benchmark dataset consists of the following steps (Fig. 1):

1. Total 1361 lncRNA entries were downloaded with curated subcellular localization from the RNAlocate database. Since multi-locational lncRNAs have multiples records in the database, we merged the entries with the same gene symbol and got 1074 unique lncRNAs;
2. We screened off the lncRNAs that do not have specific sequence information in NCBI and Ensembl, and obtained lncRNA sequence records in 7 single subcellular locations and 19 combinations of multiple subcellular locations;
3. To reduce the bias of redundant sequences on the classifiers, we used the cd-hit tool (Huang *et al.*, 2010) to remove the redundant sequences with a cutoff of 80%;
4. Since the multi-locational lncRNAs are too few to have statistical meaning, we only chose the lncRNAs that are associated to only one location for training. The remaining lncRNAs cover 7 compartments. Then we further removed two of them, i.e. endoplasmic reticulum and synapse, because they have too few samples (<10). Finally, we obtained a benchmark dataset of 612 lncRNAs, covering 5 subcellular compartments (Fig. 2). Table 1 lists the detailed statistics of the dataset.

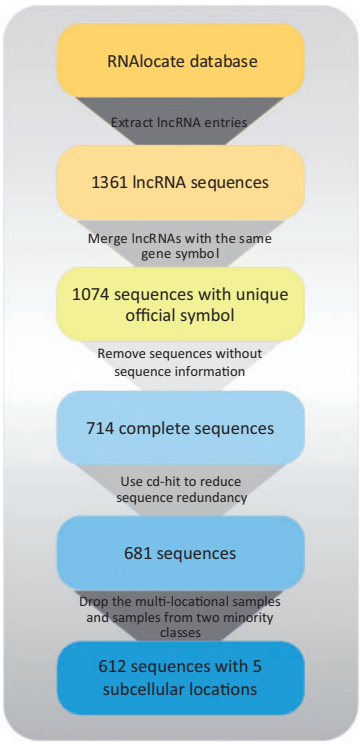

Fig. 1. The flowchart of benchmark dataset construction

2.2 Methods

The proposed lncLocator consists of three major steps, as listed in the following and shown in Figure 3.

Step 1: Feature representation. First, raw *k*-mer frequency features from the input lncRNA sequences are extracted; meanwhile an unsupervised stacked autoencoder (AE) engine is also used to learn the high-level abstractions of raw sequences. Then, both raw and high-level features are fed into the prediction engines.

Step 2: Prediction engine construction. Random forest (RF) and support vector machine (SVM) are used as the basic classification models. To reduce the effect of imbalanced distribution of training set, we apply an over-sampling technique to balance the samples among different classes.

Step 3: Stacked ensemble. Given the two types of features (Raw *k*-mers and AE-based high-level features) and two basic classifiers (RF and SVM), we then obtain a total of four base classifiers, namely RF^R, SVM^R, RF^A, SVM^A, where RF^R and SVM^R denote the models trained using the raw *k*-mer features, RF^A and SVM^A denote

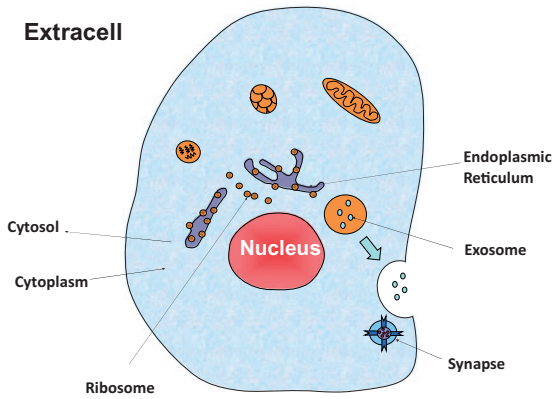

Fig. 2. Illustration of the seven subcellular locations included in the benchmark dataset

Table 1. Benchmark lncRNA subcellular localization dataset

Subcellular Locations	Before 80% cut-off	After 80% cut-off
Cytoplasm	307	301
Nucleus	153	152
Cytosol	101	91
Ribosome	47	43
Exosome	34	25
Endoplasmic reticulum[a]	9	9
Synapse[a]	1	1

[a]These two locations are not included in the following experiments due to their small sample sizes.

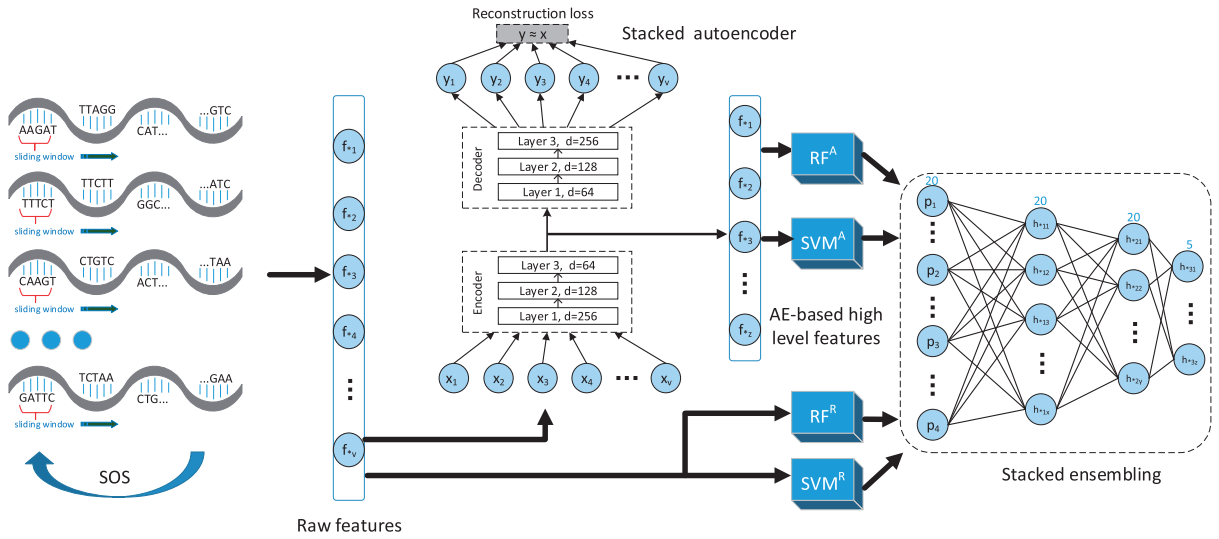

Fig. 3. The flowchart of the proposed lncLocator. RF^A and SVM^A denote the models trained using the AE-based high-level features, and RF^R and SVM^R denote the models trained using the raw *k*-mer features. SOS (Supervised Over-Sampling) is used to create new synthetic samples to balance the samples among different classes

the models trained using the AE-based high-level features. The outputs from the above 4 classifiers are integrated via a stacked ensemble model, which is the final outputs of lncLocator.

2.2.1 Feature representation

K-mer nucleotide composition features of lncRNA sequences. We extract the nucleotide composition features from lncRNA sequences. Suppose that a lncRNA sequence is represented as:

$$\mathbf{R} = \mathrm{B}_1, \mathrm{B}_2, \mathrm{B}_3, \mathrm{B}_4, \ldots, \mathrm{B}_{L-1}, \mathrm{B}_L \tag{1}$$

where B_i is one of the four nucleotide bases, A, C, G and T.

For a consecutive nucleotide segment of length k, i.e. $\{\mathrm{B}_1\mathrm{B}_2\mathrm{B}_3 \ldots \mathrm{B}_k\}$, there are 4 different choices at each of the k positions and there are 4^k different combinations of bases. Take 4-mer as an example, we have different combinations of AAAA, AAAC,$\cdots$, TTTT, and the 4-mer frequency features for RNA sequences is a 256-dimensional feature vector:

$$\mathbf{x} = [x_1, x_2, \ldots, x_{256}] \tag{2}$$

where $x_i (i = 1, 2, \ldots, 256)$ is the frequency of the ith 4-mer along the sequence.

High-level abstraction of lncRNA sequences. Compared to 20 amino acids in proteins, RNAs have a much smaller k-mer combination space, resulting in a relatively low discriminative ability of the k-mer features $\mathbf{x}$. Therefore, besides the raw k-mer features, we also adopt the unsupervised stacked autoencoder model (Vincent *et al.*, 2010) to extract the high-level abstractions from primal sequences.

Owing to the deep architecture and the reconstruction loss function, stacked autoencoders can capture high-level abstraction while keeping the core information of input data. For an original input feature $\mathbf{x}$, an encoder maps $\mathbf{x}$ to $\mathbf{y}$ with a nonlinear transform function, f,

$$\mathbf{y} = f(\mathrm{W}\mathbf{x} + b) \tag{3}$$

where W and b are two parameters to be learned.

To validate the effectiveness of the mapping, a decoder is often used to reconstruct $\mathbf{x}$ from $\mathbf{y}$ of Eq. (3):

$$\mathbf{z} = g(\mathrm{W}^T\mathbf{y} + b') \tag{4}$$

where g is also a non-linear function. In order to derive the proper mapping parameters in Eqs. (3) and (4), an optimization process is performed to minimize the loss function between $\mathbf{x}$ and $\mathbf{z}$, i.e. $\ell(\mathbf{x}, \mathbf{z})$, which is defined as the squared error function $\ell(\mathrm{x}, \mathrm{z}) = \|\mathbf{x} - \mathbf{z}\|^2$ in this study.

Based on such a single-layer autoencoder model as shown in Eqs. (3) and (4), we construct a multi-layer stacked autoencoder to generate a deep learning architecture by using the keras library (https://github.com/fchollet/keras). It is a layer-by-layer structure constructed in a sequential manner (Fig. 3). Both the encoder and decoder consist of 3 fully connected layers with dropout, where the dropout probability is set to 0.5. We feed 4-mer features into the deep model, and use 256, 128 and 64 neurons in the 3 hidden layers. A greedy layer-wise learning method is used to optimize the objective function for learning the parameters of stacked autoencoder by using the Adam optimizer.

2.2.2 Prediction engine construction

Supervised over-sampling for balancing the data distribution. As can be seen from Table 1, the numbers of lncRNA samples in different subcellular locations differ significantly. This situation may get worse in dealing with multi-class classification problems. For example, if we use the one-vs-rest strategy to discriminate the exosome lncRNAs from lncRNAs that locate at other subcellular compartment (cytoplasm, nucleus, cytosol and ribosome), the negative to positive ratio of training samples further increases to ~23:1.

The highly imbalanced data distribution significantly affects the classification performance on the minority classes since the statistical learning algorithms tend to classify the new samples to the majority classes. Many previous studies have adopted the under-sampling technique to balance the samples of different classes (Yang *et al.*, 2013; Yu *et al.*, 2014), i.e. a subset of samples is picked out from the majority class to be balanced with the minority samples size. One potential problem for using the under-sampling technique in this paper is that it will reduce the total number of lncRNA samples in the training dataset, which may also degenerate the learned classifier. Considering of these points, we extend our previous supervised over-sampling (SOS) algorithm (Hu *et al.*, 2014) from two-class classification to the multi-class classification of this study.

In contrast to the under-sampling, the over-sampling is to increase the minority class sample size by generating new synthetic samples, which results in an overall increase of the total training sample size.

Suppose $D = D^1 \cup D^2 \cup \ldots \cup D^5$ is the training dataset of 5 lncRNA subcellular locations as shown in Table 1, our purpose is to generate a new dataset $\widehat{D}$ by synthesizing new minority class samples under a supervised process. The process of our SOS method proceeds as the following:

Step 1: Initial model construction. We first train an initial classifier, denoted as random forest C_{model} on the original training dataset D:

$$\text{Training}(D) \rightarrow C_{model} \tag{5}$$

Step 2: Synthetic new sample generation. In order to synthesize a new sample for the minority class D^k, we randomly select two existing samples D_i^k and D_j^k from D^k and generate the new sample D_ξ^k as Eq. (6):

$$D_i^k + \lambda(D_i^k - D_j^k) \rightarrow D_\xi^k \tag{6}$$

where λ is a random value ranging from 0 to 1.

Step 3: Qualification test of the new sample. We apply the initial classifier C_{model} of Eq. (5) to predict D_ξ^k. If D_ξ^k is classified to D^k with a probability of $P\left(D_\xi^k\right) \in [T_{low}, T_{high}]$ by C_{model}, then D_ξ^k will be accepted otherwise, it will be rejected. In this paper, we set T_{low}=0.3 and T_{high}=1.

The above steps are repeated until the minority class has grown to twice the size it was before expansion. To illustrate the SOS idea more clearly, Figure 4 shows an example.

2.2.3 Stacked ensemble

Neural network-based ensemble decision. We use two types of features to encode the lncRNA sequences: raw *k*-mer nucleotide composition feature **x** (Eq. (2)), and the high-level feature **y** (Eq. (3)) outputted from the stacked autoencoder. Based on **x** and **y**, we trained RF and SVM classifiers, denoted as RF^R, SVM^R, RF^A, SVM^A, respectively. As expected, different classifiers have varying abilities to identify categories as shown in the following experiments. Considering the diversities of different classifiers, we construct a consensus model to enhance the prediction performance, which aims to fuse multiple base classifiers to yield higher performance. A key to achieve better performance is how to integrate different predictors. Some widely used strategies include majority voting (Breiman, 2001), averaging individual classifier results (Pan *et al.*, 2011), stacked ensemble using logistic regression (Pan *et al.*, 2016), etc.

We adopt a stacked ensemble using a 3-layer neural network (NN) to combine the prediction results from individual classifiers for the final decision. The input to the decision NN model is the outputs from RF^R, SVM^R, RF^A and SVM^A. Each individual model outputs 5 scores, indicating the probabilities of current query lncRNA belonging to the five subcellular localizations, respectively. Thus, the input layer of NN has 20 nodes, and the output layer has 5 nodes, each of which corresponds to a subcellular location class. Our final NN-based ensemble model is also implemented using keras library and scikit-learn (Pedregosa *et al.*, 2011). Figure 3 shows the flowchart of our prediction model.

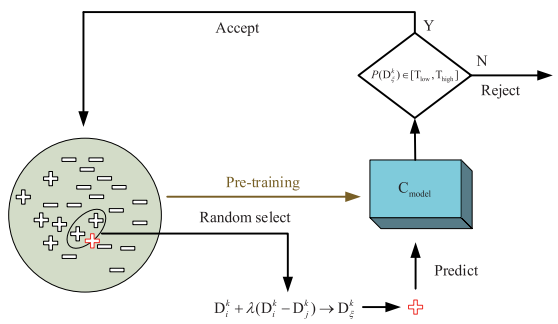

Fig. 4. The flowchart of Supervised Over-Sampling

2.2.4 Evaluation criteria

To evaluate the performance of the lncLocator model, we use accuracy, F_1 score and Recall as the evaluation criteria in our experiments through a 5-fold cross validation.

$$\text{Accuracy} = \frac{\text{Num(pred} = \text{label)}}{\text{Num(pred)}} \tag{7}$$

$$\text{Precision}^{(i)} = \frac{TP^{(i)}}{TP^{(i)} + FP^{(i)}} \tag{8}$$

$$\text{Recall}^{(i)} = \frac{TP^{(i)}}{TP^{(i)} + FN^{(i)}} \tag{9}$$

$$F_1 = \frac{1}{n}\sum_{i=1}^{n} \frac{2 \times \text{Precision}^{(i)} \times \text{Recall}^{(i)}}{\text{Precision}^{(i)} + \text{Recall}^{(i)}} \tag{10}$$

$$\text{Recall} = \frac{1}{n}\sum_{i=1}^{n} \text{Recall}^{(i)} \tag{11}$$

where $TP^{(i)}$, $FP^{(i)}$ and $FN^{(i)}$ represent true positive, false positive and false negative of the class *i*, respectively.

3 Results and discussions

3.1 Comparison between different *k*-mer frequency features for lncRNAs

We compared the performance of 4-mer, 5-mer and 6-mer frequency encoding for sequences, and the results are shown in Table 2. We did not test much higher *k*-mer features as the feature dimensions will increase exponentially. For instance, the 7-mer feature dimension is as high as 16 384, which is far beyond the number of training examples. In this case, the model has a high risk of overfitting.

An interesting phenomenon can be observed from the results: as the input dimension increases, models trained on the raw *k*-mer features and on the high-level features respond completely differently. Take RF^R and RF^A for example, on 4-mer, 5-mer and 6-mer feature sets, the F_1 scores of RF^R are 0.295, 0.275 and 0.250, respectively, showing a decreasing trend; whereas for RF^A, the performance tends to increase, i.e. 0.316, 0.323 and 0.327, respectively. The potential reason is that as *k* increases, more sequence order information is retained while more noise is introduced, then it is difficult for SVM^R to find real discriminative features from the vast number of input features, thus leading to a low generalization power. In contrast, SVM^A uses high-level abstraction features, which benefit from more abundant input information with the increasing *k*.

The other point we can see from Table 2 is that the models trained on the high level abstraction features (RF^A and SVM^A) achieve generally better performance than the models on the raw *k*-mer features (RF^R and SVM^R). This could be due to that the high-level abstraction features generated from the stacked autoencoder can grasp the hidden correlation behind high dimensional raw

Table 2. Performance of different *k*-mer frequency features for lncRNA prediction

	4-mer feature[a]			5-mer feature[a]			6-mer feature[a]		
	ACC	F_1	Recall	ACC	F_1	Recall	ACC	F_1	Recall
RF^{R} [b]	**0.575**	0.295	0.311	0.562	0.275	0.292	0.553	0.250	0.272
SVM^{R} [b]	0.534	0.226	0.253	0.506	0.155	0.212	0.501	0.146	0.207
RF^{A} [b]	0.564	**0.316**	**0.327**	**0.601**	**0.323**	**0.337**	**0.605**	0.327	0.339
SVM^{A} [b]	0.557	0.287	0.315	0.583	0.307	0.325	0.588	**0.347**	**0.356**

[a]The feature dimensions for 4-mer, 5-mer and 6-mer features are 256, 1024 and 4096, respectively.
[b]RF^{R} and SVM^{R} are trained with raw *k*-mer features, RF^{A} and SVM^{A} are trained on the high-level abstract features derived by stacked autoencoder.
The significance of bold is the maximum value of each column.

Table 3. Performance of different models on the original lncRNA benchmark dataset with 4-mer features without SOS

Method	ACC	F_1	Recall
RF^{R}	0.575	0.295	0.311
SVM^{R}	0.534	0.226	0.253
RF^{A}	0.564	0.316	0.327
SVM^{A}	0.557	0.287	0.315
LoR ensemble[a]	0.585	0.314	0.332
Average ensemble	0.588	0.310	0.328
NN ensemble[b]	**0.598**	**0.343**	**0.356**

[a]The ensemble learning using logistic regression.
[b]The ensemble learning using neural networks in lncLocator.

features, thus resulting in a higher discriminative capability. All these results demonstrate that the raw features and high-level abstraction features complement each other, and the fusion of them is expected to give a further improvement on the prediction performance.

3.2 NN ensemble decision improves the performance

In lncLocator, we applied a NN-based stacked ensemble strategy to integrate 4 predictors (RF^{R}, SVM^{R}, RF^{A}, SVM^{A}) for the final decision. We compared it with the stacked ensemble method using logistic regression (Pan *et al.*, 2016) and averaging ensemble model (Pan *et al.*, 2011). The results on 4-mer features are shown in Table 3. The stacked ensemble method was superior to the 4 individual methods and other ensemble methods on our constructed lncRNA benchmark dataset. The results indicate that our stacked ensemble is a promising approach for integrating different predictors and improving the final performance through combining diversities.

3.3 Prediction enhancement by using over-sampling technique

In previous experiments, we did not balance the training dataset. In order to alleviate the class imbalance problem, we adopted the over-sampling method, SOS. We first locally tested different models by applying the SOS approach on the 4-mer, 5-mer and 6-mer features. We found that the results on these feature encoding systems are very comparable. Considering the time cost, we choose the 4-mer features in the following experiments and as input in our final lncLocator. The results of different models on the balanced dataset (4-mer features) are shown in Table 4 and Figure 5. By comparing Tables 3 and 4, we can find that the SOS method improves the F1 score and Recall of NN ensemble from 0.343, 0.356 to 0367, 0.363, respectively, while keeping the comparable total accuracy. This is because that after balancing the dataset, the model has increased its ability for predicting the samples in the minority class. Furthermore,

Table 4. Performance of different models for classifying lncRNA on 4-mer features with class sample size balanced with SOS over-sampling

Method	ACC	F_1	Recall
RF^{R}	0.570	0.330	0.337
SVM^{R}	0.531	0.334	0.325
RF^{A}	0.572	0.353	0.354
SVM^{A}	0.527	0.349	0.347
LoR ensemble	0.580	0.328	0.335
Average ensemble	0.585	0.326	0.337
NN ensemble (lncLocator)	**0.591**	**0.367**	**0.363**

Supplementary Table S1 has provided the standard deviations of the statistical difference between different models. In addition, we also compared the ROC curves among RF^{R}, SVM^{R}, RF^{A}, SVM^{A} and NN ensemble as shown in Figure 6, where the results show that the NN ensemble method can achieve the highest AUC of 0.76.

In the imbalanced classification problem, the minority classes are often very important although they have much fewer samples than the majority classes. For instance, only a small proportion of lncRNAs are currently observed locating in exosome, and correctly predicting the samples in this class is not easy, as the trained models will give more preference to the majority classes, e.g. cytoplasm. Biologically, the functions of lncRNAs at exosome have not been well characterized, and correctly recognizing the lncRNAs of this class will give more samples for the following experimental studies. Figure 7 illustrates the confusion matrix of different models on the five subcellular location classes. As can be seen from this figure, none of samples in exosome have been correctly recognized by tested RF^{R}, SVM^{R}, RF^{A}, LoR ensemble and average ensemble, indicating developing a better predictor for accurately recognizing lncRNAs at exosome is still a very challenge future task.

3.4 A comparison to homology-transfer baseline method

The homology-transfer based approach can be considered as a nearest neighbor predictor, where the distance between two lncRNAs is measured by their sequence identity. Here, we used the blastn in the blast+ toolbox (Camacho *et al.*, 2009) to search each test sequence against the training dataset through the same five-fold cross-validation protocol. The subcellular localization of the query sequence is decided by the localization of the sequence with the lowest E-value in the training dataset. Our results show that the average accuracy, F_1-score and recall of the homology-based method are 0.493, 0.339 and 0.338, respectively, which are lower than the proposed lncLocator method. These results indicate that when the experimentally verified training samples are not enough, the homology-transfer

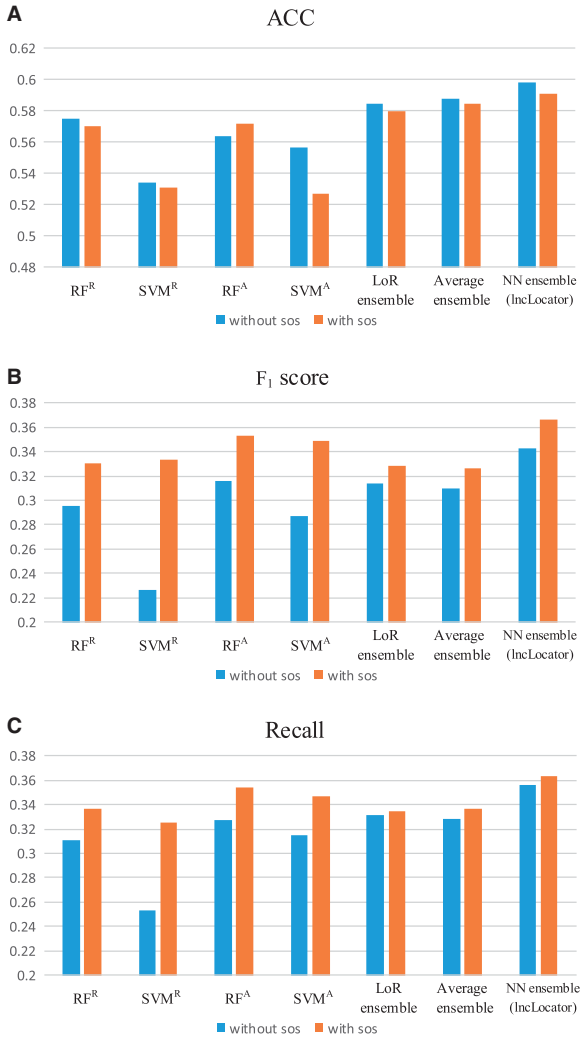

Fig. 5. The performance comparisons of different models between with-SOS (dataset with minority class samples over-sampling) and without-SOS (original imbalanced dataset). (**A**): accuracy, (**B**): F_1 score and (**C**): Recall

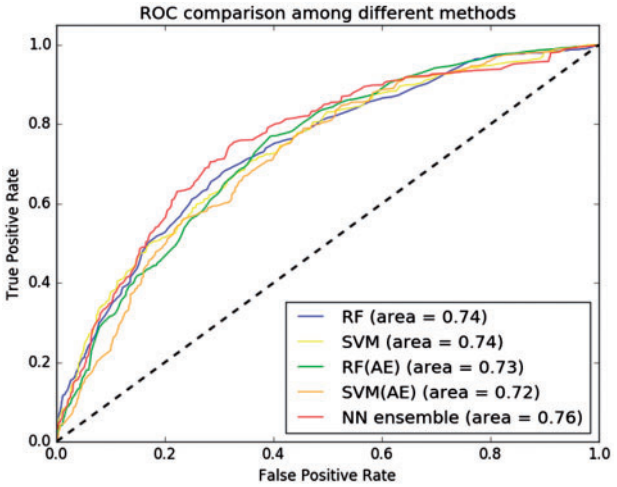

Fig. 6. ROC comparison among different methods. RF, SVM, RF(AE), SVM(AE) represent RF^R, SVM^R, RF^A, SVM^A, respectively

approach will hard to get satisfactory results. The machine learning-based model shows a more powerful performance in the lncRNA subcellular localization prediction at current stage.

3.5 The performance through an independent test protocol

Five-fold cross-validation is applied in our previous experiments. To better evaluate the performance of our model, we also tested it through an independent test protocol. We constructed a stand-alone test dataset containing 40 sequences (Supplementary Table S2), and the remaining 572 sequences used as the entire training set. The results on the test dataset are shown in Table 5, which are similar to the five-fold cross validation protocol on F_1 and recall metrics. The final NN ensemble model yields the best performance among tested approaches.

3.6 Discussions

Accurate prediction of the subcellular locations of lncRNAs is a much more challenging problem than prediction of protein subcellular locations due to there are only four states of the nucleotides. Besides, features directly extracted from the nucleotide sequences will also be biased from probable mutations, resulting in a potential noise effects in the features. In this study, we propose to use unsupervised stacked autoencoder to extract high-level abstractions of the *k*-mer features, which is demonstrated significantly helpful for enhancing the subsequent classification. The reason is that deep learning architecture is capable to learn complicate statistical characteristics hidden in the raw data (Zhou and Troyanskaya, 2015). The *k*-mer frequency features are similar to words in the document, and deep learning model can extract the high-level abstractions like topics in articles. The compressed representation also avoids the curse of dimensionality via eliminating irrelevant variabilities, especially for high-dimensional *k*-mer frequency features. The improved performance on the consensus model by fusing the raw features and high-level abstractions also demonstrate that these two types of feature encoding systems complement each other very well.

Different predictors have their own advantages. Specifically, no single method can surpass others in all respects. In this study, the stacked NN ensemble method is designed to integrate different models in lncLocator. Different from average voting or majority voting, the stacked NN ensemble can combine the strengths of individual predictors with automatic weight learning. We also show that the NN model is also superior to the widely-used logistic regression ensemble approach in this study.

Classification for the minority lncRNA class (e.g. exosome) is a particular challenging problem in this study. As shown in Figure 7, none of samples in exosome have been correctly recognized by tested RF^R, SVM^R, RF^A, LoR ensemble and average ensemble. This could be due to that the pattern of lncRNA samples in exsome is very similar to other classes and also because there are too few samples in exosome. The statistically learned model will naturally give more preference to the majority class, resulting in a very bad performance for the minority class. In this study, we have proposed the SOS algorithm for generating some synthetic samples in the minority classes (e.g. exosome) to balance the dataset distribution. Our results show that this could be a promising strategy, although much work is still needed to further enhance the minority class classification performance.

We also analyzed the impact of sequence similarity on the predictive performance, and tested the models with sequences redundancy at different cutoff values. Beside the 80% cutoff value tested above, the results of other cut-offs of 50, 60 and 70% are shown in

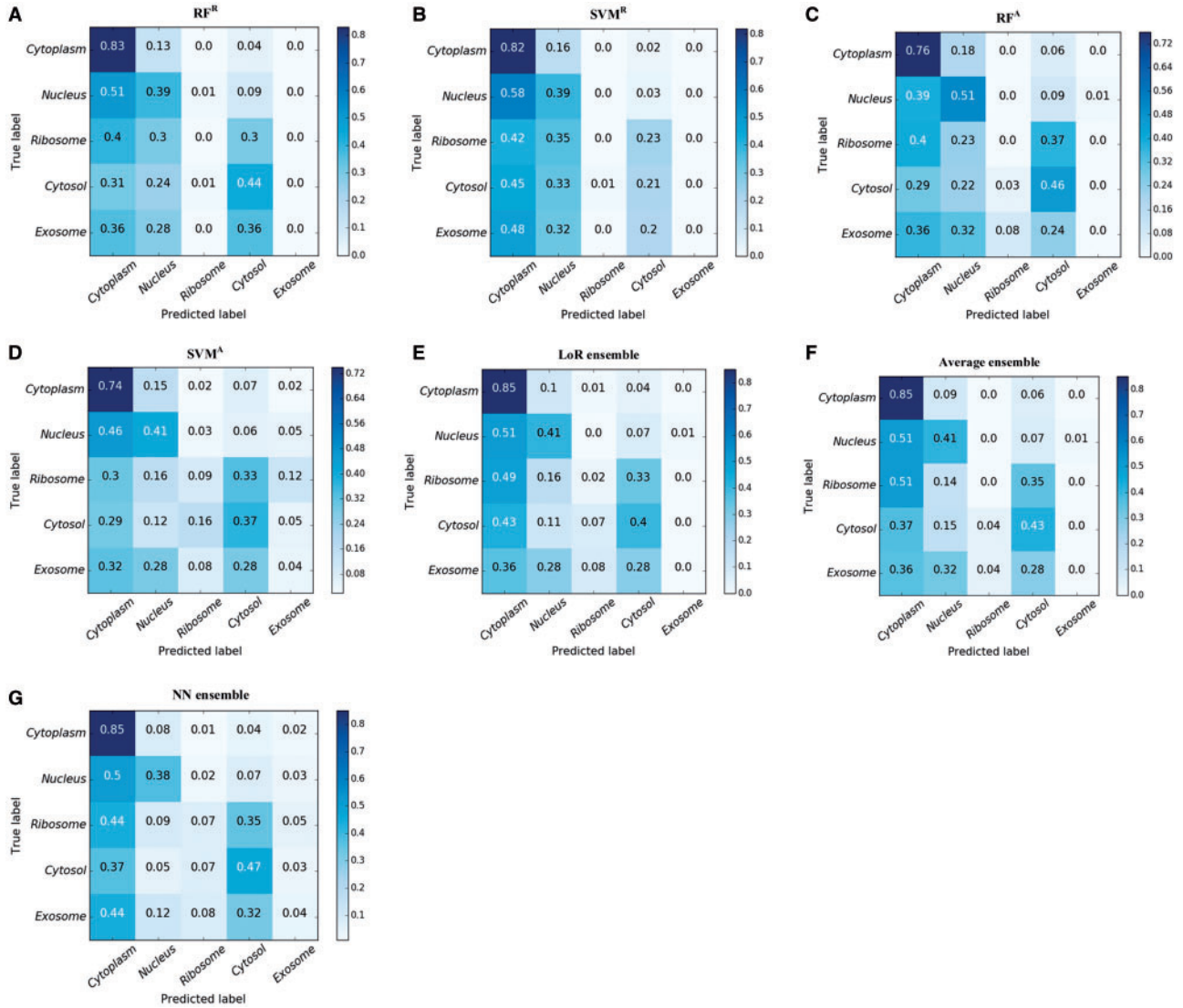

Fig. 7. Confusion matrix of the lncRNA classification with SOS. (**A**): RF^R, (**B**) SVM^R, (**C**) RF^A, (**D**) SVM^A, (**E**) LoR ensemble, (**F**) Average ensemble and (**G**) NN ensemble (lncLocator)

Table 5. Performance of different models for classifying lncRNA on 4-mer features using SOS over-sampling on the stand-alone test dataset

Method	ACC	F_1	Recall
RF^R	0.425	0.289	0.340
SVM^R	0.375	0.249	0.300
RF^A	0.400	0.276	0.320
SVM^A	0.375	0.254	0.300
LoR ensemble	0.425	0.333	0.360
Average ensemble	0.425	0.294	0.340
NN ensemble (lncLocator)	**0.450**	**0.351**	**0.380**

Supplementary Tables S3–S6. On all the tested sets, the final NN-ensemble model is superior to other tested methods in most cases. The benchmark dataset with 50% cutoff yields the lowest F_1-score and recall, probably due to the lower sequence similarity and fewer number of training samples.

Our results show that the prediction performance will be affected by the sequence length. The length of lncRNAs in our constructed dataset ranges from 192 to 91 671. We divided it into 5 intervals [192, 1000), [1000, 2000), [2000, 3000), [3000, 5000) and [5000, 91 671]. The corresponding accuracies in the 5 intervals are 0.521, 0.654, 0.612, 0.598 and 0.463, respectively (Supplementary Table S7). The longest sequences have relatively lower accuracies, while the sequences in [1000, 2000) have the highest accuracy. For the shorter sequences with length in [192, 1000], they may contain insufficient information for accurate prediction compared to sequences with medium length in [1000, 5000); while for the sequences in [5000, 91 671], they could be too long for the classifier to extract high quality features.

Although the lncLocator automatically extracts high level features using deep neural networks and those learned high-level features show higher discrimination power, these features are still not well explained from the biological perspectives. In future work, we will explore better network architectures to learn high level features with biological insights. In addition, the over-sampling strategy has been used to alleviate the imbalance problem here and our results show that the performance is affected by the sampling ratios (Supplementary Table S8). With the progress of RNA annotations,

we expect to collect more labeled samples and expand the benchmark dataset to train a more powerful model. The features we currently used in lncLocator is only *k*-mer frequency features, there are many other useful features that could be integrated for better identifying the subcellular location. For instance, GO features and secondary structure information features.

4 Conclusion

In this study, to the best of our knowledge, we present the first computational method lncLocator to predict lncRNA subcellular localization, which is an ab initio approach only requiring the nucleotide sequences as inputs. We have designed the unsupervised deep stacked architecture to extract high-level abstraction features and integrate the outputs from different models. Our results have demonstrated the efficacy of the ensemble model. Considering the data imbalance in this study, we propose to use the over-sampling method to improve the performance of the model, without reducing the total sample size in the dataset. A future challenge is to explore a better way to further improve the prediction performance in the minority classes. We plan to mine more biologically targeting motifs specifically for these locations by designing better network architectures.

Acknowledgements

We thank Marten van den Berg for proof reading this manuscript.

Funding

This work was supported by the National Natural Science Foundation of China (No. 61725302, 61671288, 91530321, 61603161) and Science and Technology Commission of Shanghai Municipality (No. 16JC1404300, 17JC1403500, 16ZR1448700).

Conflict of Interest: none declared.

References

Ayers,D. (2013) Long non-coding RNAs: novel emergent biomarkers for cancer diagnostics. *J. Cancer Res. Treat.*, **1**, 31–35.

Batista,P.J. and Chang,H.Y. (2013) Long noncoding RNAs: cellular address codes in development and disease. *Cell*, **152**, 1298–1307.

Breiman,L. (2001) Random forests. *Mach. Learn.*, **45**, 5–32.

Brennecke,J. *et al.* (2005) Principles of microRNA–target recognition. *PLoS Biol.*, **3**, e85.

Cabili,M.N. *et al.* (2015) Localization and abundance analysis of human lncRNAs at single-cell and single-molecule resolution. *Genome Biol.*, **16**, 20.

Camacho,C. *et al.* (2009) BLAST+: architecture and applications. *BMC Bioinformatics*, **10**, , 421.

Chawla,N.V. *et al.* (2002) SMOTE: synthetic minority over-sampling technique. *J. Artif. Intell. Res.*, **16**, 321–357.

Chen,L. (2016) Linking long noncoding RNA localization and function. *Trends Biochem. Sci.*, **41**, 761–772.

Chen,L. and Carmichael,G.G. (2010) Decoding the function of nuclear long non-coding RNAs. *Curr. Opin. Cell Biol.*, **22**, 357–364.

Chou,K.C. and Shen,H.B. (2008) Cell-PLoc: a package of Web servers for predicting. *Nat. Protoc.*, **3**, 153–162.

Chou,K. and Cai,Y. (2003) A new hybrid approach to predict subcellular localization of proteins by incorporating gene ontology. *Biochem. Biophys. Res. Commun.*, **311**, 743–747.

Di Lena,P. *et al.* (2012) Deep architectures for protein contact map prediction. *Bioinformatics*, **28**, 2449–2457.

Fan,X. and Zhang,S. (2015) lncRNA-MFDL: identification of human long non-coding RNAs by fusing multiple features and using deep learning. *Mol. BioSyst.*, **11**, 892–897.

Heffernan,R. *et al.* (2015) Improving prediction of secondary structure, local backbone angles, and solvent accessible surface area of proteins by iterative deep learning. *Sci. Rep.*, **5**, 11476.

Hu,J. *et al.* (2014) A new supervised over-sampling algorithm with application to protein-nucleotide binding residue prediction. *PloS One*, **9**, e107676.

Huang,Y. *et al.* (2010) CD-HIT Suite: a web server for clustering and comparing biological sequences. *Bioinformatics*, **26**, 680–682.

Iyer,M.K. *et al.* (2015) The landscape of long noncoding RNAs in the human transcriptome. *Nat. Genet.*, **47**, 199–208.

Lecun,Y. *et al.* (2015) Deep learning. *Nature*, **521**, 436–444.

Li,Y. *et al.* (2013) HMDD v2. 0: a database for experimentally supported human microRNA and disease associations. *Nucleic Acids Res.*, **42**, D1070–D1074.

Marchler-Bauer,A. *et al.* (2005) CDD: a Conserved Domain Database for protein classification. *Nucleic Acids Res.*, **33**, D192–D196.

Mattick,J.S. and Makunin,I.V. (2006) Non-coding RNA. *Hum. Mol. Genet.*, **15**, R17.

Min,S. *et al.* (2017) Deep learning in bioinformatics. *Brief. Bioinf.*, **18**, 851–869.

Nair,R. and Rost,B. (2002) Sequence conserved for subcellular localization. *Protein Sci.*, **11**, 2836–2847.

Pan,X. and Shen,H. (2017) RNA-protein binding motifs mining with a new hybrid deep learning based cross-domain knowledge integration approach. *BMC Bioinformatics*, **18**, 136.

Pan,X. *et al.* (2016) IPMiner: hidden ncRNA–protein interaction sequential pattern mining with stacked autoencoder for accurate computational prediction. *BMC Genomics*, **17**, 582.

Pan,X. *et al.* (2011) Towards better accuracy for missing value estimation of epistatic miniarray profiling data by a novel ensemble approach. *Genomics*, **97**, 257–264.

Park,K. and Kanehisa,M. (2003) Prediction of protein subcellular locations by support vector machines using compositions of amino acids and amino acid pairs. *Bioinformatics*, **19**, 1656–1663.

Pedregosa,F. *et al.* (2011) Scikit-learn: machine learning in Python. *J. Mach. Learn. Res.*, **12**, 2825–2830.

Pierleoni,A. *et al.* (2011) MemLoci: predicting subcellular localization of membrane proteins in eukaryotes. *Bioinformatics*, **27**, 1224–1230.

Savojardo,C. *et al.* (2015) TPpred3 detects and discriminates mitochondrial and chloroplastic targeting peptides in eukaryotic proteins. *Bioinformatics*, **31**, 3269–3275.

Shen,H. and Chou,K. (2006) Ensemble classifier for protein fold pattern recognition. *Bioinformatics*, **22**, 1717–1722.

Shen,H. and Chou,K. (2007) Hum-mPLoc: an ensemble classifier for large-scale human protein subcellular location prediction by incorporating samples with multiple sites. *Biochem. Biophys. Res. Commun.*, **355**, 1006–1011.

Shen,H. and Chou,K. (2008) PseAAC: a flexible web server for generating various kinds of protein pseudo amino acid composition. *Anal. Biochem.*, **373**, 386–388.

Shen,H. and Chou,K. (2009) A top-down approach to enhance the power of predicting human protein subcellular localization: hum-mPLoc 2.0. *Anal. Biochem.*, **394**, 269–274.

Spencer,M. *et al.* (2015) A deep learning network approach to ab initio protein secondary structure prediction. *IEEE/ACM Trans. Comput. Biol. Bioinf.*, **12**, 103–112.

Thomson,J.M. *et al.* (2004) A custom microarray platform for analysis of microRNA gene expression. *Nat. Methods*, **1**, 47–53.

Vincent,P. *et al.* (2010) Stacked denoising autoencoders: learning useful representations in a deep network with a local denoising criterion. *J. Mach. Learn. Res.*, **11**, 3371–3408.

Vlachos,I.S. *et al.* (2012) DIANA miRPath v. 2.0: investigating the combinatorial effect of microRNAs in pathways. *Nucleic Acids Res.*, **40**, W498–W504.

Wan,S. *et al*. (2017) FUEL-mLoc: feature-unified prediction and explanation of multi-localization of cellular proteins in multiple organisms. *Bioinformatics*, **33**, 749–750.

Yang,J. *et al*. (2013) High-accuracy prediction of transmembrane inter-helix contacts and application to GPCR 3D structure modeling. *Bioinformatics*, **29**, 2579–2587.

Yu,D. *et al*. (2014) Enhancing protein-vitamin binding residues prediction by multiple heterogeneous subspace SVMs ensemble. *BMC Bioinformatics*, **15**, 297.

Zhang,T. *et al*. (2017) RNALocate: a resource for RNA subcellular localizations. *Nucleic Acids Res*., **45**, D135–D138.

Zhou,H. *et al*. (2017) Hum-mPLoc 3.0: prediction enhancement of human protein subcellular localization through modeling the hidden correlations of gene ontology and functional domain features. *Bioinformatics*, **33**, 843–853.

Zhou,J. and Troyanskaya,O.G. (2015) Predicting effects of noncoding variants with deep learning-based sequence model. *Nat. Methods*, **12**, 931–934.