BMC Genomics

RESEARCH Open Access

Deep neural networks for inferring binding sites of RNA-binding proteins by using distributed representations of RNA primary sequence and secondary structure

Lei Deng[1], Youzhi Liu[1], Yechuan Shi[1], Wenhao Zhang[2], Chun Yang[3]* and Hui Liu[2]*

From 2019 IEEE International Conference on Bioinformatics and Biomedicine (IEEE BIBM 2019)
San Diego, CA, USA. 18–21 November 2019

Abstract

Background: RNA binding proteins (RBPs) play a vital role in post-transcriptional processes in all eukaryotes, such as splicing regulation, mRNA transport, and modulation of mRNA translation and decay. The identification of RBP binding sites is a crucial step in understanding the biological mechanism of post-transcriptional gene regulation. However, the determination of RBP binding sites on a large scale is a challenging task due to high cost of biochemical assays. Quite a number of studies have exploited machine learning methods to predict binding sites. Especially, deep learning is increasingly used in the bioinformatics field by virtue of its ability to learn generalized representations from DNA and protein sequences.

Results: In this paper, we implemented a novel deep neural network model, DeepRKE, which combines primary RNA sequence and secondary structure information to effectively predict RBP binding sites. Specifically, we used word embedding algorithm to extract features of RNA sequences and secondary structures, i.e., distributed representation of k-mers sequence rather than traditional one-hot encoding. The distributed representations are taken as input of convolutional neural networks (CNN) and bidirectional long-term short-term memory networks (BiLSTM) to identify RBP binding sites. Our results show that deepRKE outperforms existing counterpart methods on two large-scale benchmark datasets.

Conclusions: Our extensive experimental results show that DeepRKE is an efficacious tool for predicting RBP binding sites. The distributed representations of RNA sequences and secondary structures can effectively detect the latent relationship and similarity between k-mers, and thus improve the predictive performance. The source code of DeepRKE is available at https://github.com/youzhiliu/DeepRKE/.

Keywords: RNA-binding proteins, Binding sites, Distributed representation, k-mer, Deep learning, Convolutional neural network, Bidirectional long short term memory network

*Correspondence: yangchun119119@163.com; hliu@cczu.edu.cn
[2]Aliyun School of Big Data, Changzhou University, 213164, Changzhou, China
[3]Department of Obstetrics, The Affiliated Changzhou No.2 People's Hospital of Nanjing Medical University, Changzhou, China
Full list of author information is available at the end of the article

Background

RNA-binding proteins (RBPs) have important functions in many biological processes, e.g. gene regulation, RNA editing, mRNA processing and other biological processes [1, 2]. It is estimated that 6% -8% of the proteins in eukaryotes are RNA binding proteins, but so far only a few RNA binding proteins (HuR, AUF1, TTP, TIA1, CUBBP2, etc) have been approved to be specifically involved in mRNA stability, translation and other levels of gene regulation [3–7]. Therefore, the identification of RBP binding sites is crucial to understanding the mechanism of biological processes. Recently, various of high-throughput biochemical methods have been proposed to study and analyze the proteins-RNA complexes to identify the binding sites of RNA molecules, among which the popular ones are CLIP-Seq [8], RNACompete [9], eCLIP [10] and PAR-CLIP [11]. However, these techniques still cost-heavy and time-intensive, which hampers the exploration of RBP binding sites.

With the increasing volume of verified RBP binding sites, quite a few studies focused on developing computational prediction models based on the known RBP binding sites. As is shown in Table 1, these methods mostly employ RNA sequence and structural information to predict protein-RNA interactions. For instance, RNAcontext [12] uses position weight matrix (PWM) of RNA sequence and secondary structure profile to predict the binding preference of RBP. RCK [13] is an extension of RNAcontext, which uses a novel k-mer-based model to further improve the predictive performance. GraphProt [14] adopts the form of graph coding to integrate RNA sequence and structure into the graph kernel to generate a feature vector of more than 3,000 dimensions, which is subsequently used as the input of an SVM model to predict RBP binding preference. IONMF [15] proposes a feature representation method of orthogonal matrix eigen-decomposition, which integrates the k-mer sequence, secondary structure, gene ontology (GO) information and region type as input into a machine learning model to predict binding sites. Oli [16] uses k-mer frequency as input feature into an SVM classifier to predict RNA-protein interactions. Rather than commonly constructing a binary classification task, RNAcommender [17] adopts recommendation system to prioritize RNA against undeveloped RNA binding proteins by disseminating available interaction information based on protein domain composition and RNA predicted secondary structure.

Table 1 Computational methods for RBP binding preference prediction

Method	Sequence	Structure	Reference
RNAcontext	Yes	Yes	[12]
GraphProt	Yes	Yes	[14]
iONMF	Yes	Yes	[15]
Oli	Yes	Yes	[16]
RNAcommender	Yes	Yes	[17]
RCK	Yes	Yes	[13]
DeepBind	Yes	No	[18]
Deepnet-rbp	Yes	No	[19]
DanQ	Yes	No	[20]
iDeepS	Yes	Yes	[21]
iDeepV	Yes	No	[22]
Pysster	Yes	Yes	[23]
DLPRB	Yes	Yes	[24]

"Yes" and "No" means whether the computational methods uses sequence and structure information to predict the binding site

Recently, deep learning [25] has achieved remarkable success in many fields, such as image processing [26], natural language processing [27] and speech recognition [28], and thus drawn greater attention from bioinformaticians [29]. Through multiple hidden layers that perform feature transformation in the deep neural network, the feature representation in the original space is mapped into a new feature space, which makes the task of interest smoother. Based on the set of verified RBP binding sites, more and more studies use deep learning to predict RBP binding sites. When constructing the input of the neural network, most methods adopt one-hot coding, e.g., the four nucleotides A, C, G, and T are encoded as (0, 0, 0, 1), (0, 1, 0, 0), (0, 0, 1, 0) and (0, 0, 0, 1), and thus a RNA sequence of length L is transformed into a matrix of size $4*L$. DeepBind [18] is the first to use CNN (Convolutional Neural Network) [30] to predict protein-DNA/RNA binding preferences based on one-hot coding of nucleotide sequences. DanQ [20] and iDeepS [21] also adopt one-hot encoding of nucleotide sequences as input of deep learning models to predict protein-RNA binding preferences. Of note, iDeepS additionally makes use of the information of RNA secondary structure profiles. Pysster [23] exploits a novel strategy to expand the one-hot coding to predict protein-RNA binding preferences. It combines RNA sequence and secondary structure alphabet into an extended alphabet composed of arbitrary characters, which integrates RNA Sequence and structure input strings. Then, the proposed string is encoded as one-hot and used as the input. However, a fatal disadvantage of traditional one-hot coding is the problem of curse of dimensionality. Accordingly, deep-rbps [19] uses k-mer frequency coding to encode sequence, secondary structure and tertiary structure into a unified feature representation, which is subsequently fed into a multi-modal DBN [31] to predict RBP binding sites and motifs. In general, k-mer frequency coding greatly reduces the dimension of the input of deep neural network. Taking 4-mer peptide for example, in all possible cases, only a total of 256 kinds of 4-mer (AAAA, AAAC ... TTTT) peptide are required to be counted the frequencies included in the sequence, resulting to a 256-D final vector. While k-mer frequency coding can effectively reduce the dimension, it ignores the position information of the sequence that is actually important for the prediction of RBP sites. Moreover, k-mer frequency coding does not consider contextual correlation. Inspired by the field of natural language processing, DNA2vec [32], BioVec [33], seq2vec [34], and Gene2Vec [35] heuristically use word2vec [36] to obtain a distributed representation from genomic sequences without supervision. iDeepV [22] applies the word embedding method to learn distributed representation of k-mers, and greatly improves the prediction accuracy.

In this paper, we present a novel deep neural network, DeepRKE, which consists of CNN and bidirectional LSTM, to infer latent RBP binding sites. We used the word embedding model to build distributed representations of RNA sequences and secondary structures, and input them into a deep neural network to predict RBP binding sites. The skip-gram algorithm [37] uses the input of current word to predict the surrounding context words, and can effectively capture the contextual information. Thereby, we used skip-gram algorithm to learn a k-dimensional distributed representation of RNA sequence and RNA secondary sequence in a low-dimensional space. DeepRKE has the following contributions: 1) k-mer embedding is used to represent both RNA sequence and RNA secondary structure instead of traditional one-hot encoding. 2) We use three CNN modules, two modules are used to extract the features of RNA sequence and secondary structure respectively, and the third module is used to capture the relationship between sequence and structure. 3) DeepRKE can handle the sequences with variable length. We evaluated DeepRKE on two large-scale benchmark datasets, and also assessed its performance on sequences with fixed length and variable length. The experimental results demonstrated that deepRKE achieved better performance than five competitive methods.

Results

In this part, First we evaluated DeepRKE with other five state-of-the-art methods on two large-scale benchmark datasets RBP-24 [13] and RBP-31 [38]. Next, We deleted the secondary structure profile from the input, and then only used the sequence as the input of DeepRKE to judge the impact of RNA secondary structure on the prediction ability from the final prediction effect. In addition, we also evaluated the performance improvement of Deep- RKE by using distributed representation or not, as well as one-hot encoding. Finally, for more insights into the performance improvement by BLSTM, we compare DeepRKE with a variant using only CNNs without BLSTM layer.

DeepRKE learning framework

We implemented the learning framework of DeepRKE to infer RNA-proteins binding site on RNAs, as shown in Fig. 1. First, we used RNAShapes [39] to predict the RNA secondary structure. Second, we used the word2vec algorithm to learn distributed representation of 3-mers of RNA sequences and secondary structure sequences. The distributed representations were used as the input of two CNNs (one is for RNA sequence and the other is for secondary structure) to transform features of sequences and structure, respectively. Next, we combined the output features and fed them into another CNN, which was followed by a bidirectional LSTM and two fully connected layers. Finally, a sigmoid function was used to predict the probability of RBP binding sites.

Performance comparison to counterpart methods

We evaluated DeepRKE with other multiple existing methods on two large-scale benchmark datasets, including RBP-24 [14] and RBP-31 [40]. The RBP-24 dataset contains RNA sequences with variable length ranging from 150 to 375, while the RBP-31 dataset contains fixed-length RNA sequences with 101 nucleotides. The counterpart methods are GraphProt, deepnet-rbp, DeepBind, iDeepS and iDeepV. Note that iDeepV and DeepBind use only sequences, while DeepRKE, iDeepS and GraphProt use both sequences and secondary structures. DeepBind and iDeepS use one-hot encoding to represent RNA sequence, DeepRKE and iDeepV use distributed representations of RNA sequence.

First, we conducted the performance evaluation on the RBP-24 dataset with GraphProt, deepnet-rbp, DeepBind and iDeepV that can handle variable sequences. Table 2 shows the number of training and test samples gained by each biochemical assay included in the RBP-24 dataset, as well as the corresponding AUC performance measures (detailed results are shown in Additional file 1). It can be found that among the 24 RBPs, the average AUC of DeepRKE is 0.934, higher than that of the secondary method DeepBind which is 0.917. GraphProt obtained the worst performance 0.887. More importantly, DeepRKE obtains the best AUCs on 18 out of total 24 sets of RBPs. It is worth noting that the performance of the four deep learning-based methods perform far more superior to the GraphProt that is traditional machine learning-based. iDeepV adopts the distributed representation, but merely uses RNA sequences. As a result, its performance is better than GraphProt and deepnet-rbp, but worse than DeepRKE. For example, on ALKBH5 and C17ORF85, DeepRKE obtains the AUC values of 0.740 and 0.824, which is an increase of 15% and 11% compared to the AUCs of 0.643 and 0.74 achieved by iDeepV, respectively. For some RBPs,

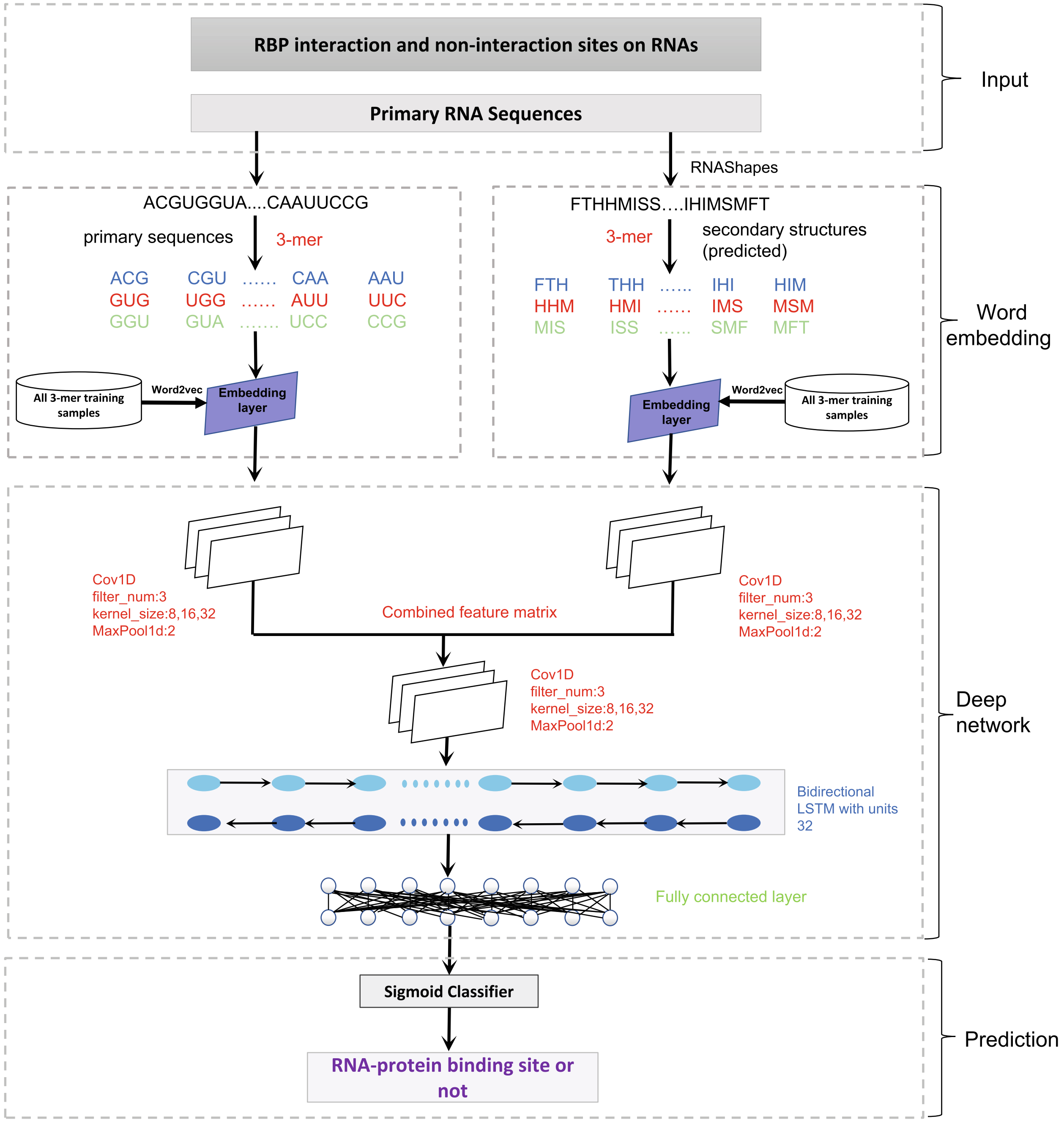

Fig. 1 The illustrative flowchart of DeepRKE learning framework. First, we use RNAShapes to predict the RNA secondary structure from primary sequences. Second, word embedding algorithm is used to learn the distributed representations of 3-mers from primary sequences and secondary structures. Third, the learned distributed representations are fed into two CNNs (one is for RNA sequence and the other for secondary structures) to transform sequence and structure features, which are in turn input into a CNN module and a bidirectional LSTM layer followed by two fully connected layer. Finally, we use a sigmoid classifier to predict the probability of being RBP binding sites

DeepRKE greatly boost the prediction performance, e.g. DeepRKE increases the AUC values by 18% and 14% compared to GraphProt on Ago2 and TIAL1.

On RBP-31 dataset, we also compared DeepRKE to iDeepS and Oli's method [16]. The AUC values of each method are shown in Additional file 2. The average AUCs of six competitive methods on 31 set of RBPs are illustrated in Fig. 2, where the performance of DeepRKE is significantly superior to other 5 methods with the average AUC of 0.873.

Table 2 Performance comparison between DeepRKE, GraphProt, deepnet-rbp, DeepBind and iDeepV on RBP-24 dataset

RBP	#positives	#negatives	GraphProt	deepnet-rbp	DeepBind	iDeepV	DeepRKE
ALKBH5 PAR-CLIP	1213	1197	0.680	0.714	0.668	0.643	**0.740**
C17ORF85 PAR-CLIP	1860	1849	0.800	0.820	0.755	0.740	**0.824**
C22ORF28 PAR-CLIP	9369	9136	0.751	0.792	0.809	0.823	**0.832**
CAPRIN1 PAR-CLIP	8140	7901	0.855	0.834	**0.888**	0.824	0.869
Ago2 HITS-CLIP	48,095	44,251	0.765	0.809	0.879	0.886	**0.900**
ELAVL1 HITS-CLIP	8595	8436	0.955	0.966	**0.980**	0.966	0.978
SFRS1 HITS-CLIP	19,438	17,195	0.898	0.931	0.929	0.905	**0.945**
HNRNPC iCLIP	21,472	19,794	0.952	0.962	**0.979**	**0.979**	0.978
TDP43 iCLIP	92,031	75,079	0.874	0.876	0.930	0.935	**0.954**
TIA1 iCLIP	18,049	16,135	0.861	0.891	0.929	0.941	**0.942**
TIAL1 iCLIP	42,332	36,652	0.833	0.870	0.922	0.929	**0.946**
Ago1-4 PAR-CLIP	36,902	31,310	0.895	0.881	0.919	0.925	**0.932**
ELAVL1 PAR-CLIP(B)	9464	9283	0.935	0.961	0.961	0.962	**0.980**
ELAVL1 PAR-CLIP (A)	27,275	23,974	0.959	0.966	0.972	0.973	**0.978**
EWSR1 PAR-CLIP	16,292	14,720	0.935	0.966	0.969	0.962	**0.971**
FUS PAR-CLIP	34,581	31,480	0.968	0.980	0.983	0.976	**0.988**
ELAVL1 PAR-CLIP(C)	125,202	113,686	0.991	0.994	0.989	0.990	**0.996**
IGF2BP1-3 PAR-CLIP	8539	6838	0.889	0.879	0.939	0.923	**0.943**
MOV10 PAR-CLIP	13,793	12,987	0.863	0.854	0.899	0.896	**0.920**
PUM2 PAR-CLIP	9116	8227	0.954	**0.971**	0.964	0.965	0.965
QKI PAR-CLIP	10,276	9142	0.957	**0.983**	0.973	0.965	0.975
TAF15 PAR-CLIP	7298	6606	0.970	0.983	0.978	0.978	**0.985**
PTB HITS-CLIP	44,574	43,700	0.937	**0.983**	0.944	0.936	0.953
ZC3H7B PAR-CLIP	20,962	20,018	0.820	0.796	0.875	0.883	0.914
Mean AUC			0.887	0.902	0.917	0.913	0.934

Note: boldface is the best experimental results for this experiment

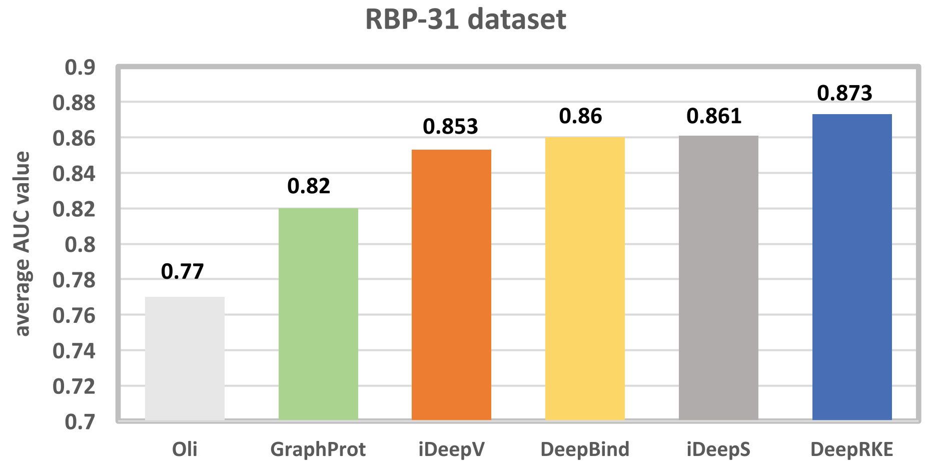

Fig. 2 Performance comparison between DeepRKE, iDeepV, iDeepS, DeepBind and GraphProt on RBP-31 dataset. All methods are run on the same training and independent test set across 31 set of RBPs (x-axis)

Note that the current state-of-the-art method iDeepS gets an average AUC of 0.86, which is still slightly worse than DeepRKE. In fact, our method performs better than iDeepS on quite a few sets of RBPs. For example, DeepRKE increases the AUC by 8.2%, 5.1%, 5%, 11%, 7.5% on the Ago/EIF, Aog2-MNase, Ago2-1, IGFBP1-3 and MOV10, respectively. Moreover, we compared DeepRKE with GraphProt, which is a structure profile-based method and demonstrates better performance than RNAcontext [12]. GraphProt has the average AUC of 0.82, which is inferior to 0.873 of DeepRKE. Remarkably, DeepRKE achieves better AUCs than GraphProt in all experiments. The average AUC of Oli is 0.77, which is significantly lower than DeepRKE. Oli even obtains the performance close to random guessing on some sets of RBPs, e.g. on Ago2-MNase its AUC is only 0.512.

In summary, DeepRKE achieves significant performance on both RBP-24 and RBP-31 datasets. DeepRKE not only outperforms the current methods for predicting RBP binding sites, but also successfully handles input sequences with variable length, which fails to be addressed by iDeepV and iDeepS.

RNA secondary structure impacts predictive power

The results in Table 2 has somewhat demonstrated the importance of RNA secondary structure, e.g., the methods that adopts the secondary structure information commonly outperform those use only sequence information. The proposed deep learning framework allows us to more strictly investigate the impact of RNA structure on performance, namely, whether and what extend the secondary structure can contribute to the prediction of RBP binding site. Accordingly, we removed the secondary structural profiles from input, and then took only the sequence as the input of our model. For clarity, we referred to this simplified model as DeepRKE-, indicating that RNA structure information is not taken into account.

As shown in Table 1 and Fig. 3a, DeepRKE- achieves an average AUC of 0.924 on RBP-24 dataset, which is lower that of DeepRKE 0.934. We observed significant decrease in performance (with AUC score reduced by >1%) in eight RBPs, i.e. ALKBH5, C17ORF85, CAPRIN1, SFRS1, TIA1, ELAVL1(B), MOV10, ZC3H7B. Especially, on the two proteins ALKBH5 and CAPRIN1, AUC decreases by 3% and 6%, respectively. As shown in Fig. 3b, on RBP-31 dataset, DeepRKE- achieves an average AUC of 0.863 over 31 sets of RBPs, which is still worse than that of 0.873 achieved by DeepRKE. On the two proteins Ago2-MNase and ELAVL1-MNase, AUC decreases by 6% and 5%, respectively. The result implies that RNA secondary structure significantly contributes to the prediction of RBP binding sites.

Distributed representation significantly improves performance

Rather than traditional one-hot encoding, DeepRKE adopts distributed representations of RNA sequence to capture the high-order dependencies among nucleotides, leading to dimension-reduced feature vectors. To justify the advantage of distributed representation in identifying binding site of RNA-binding proteins, we removed the secondary structural profiles (as it can not be represented

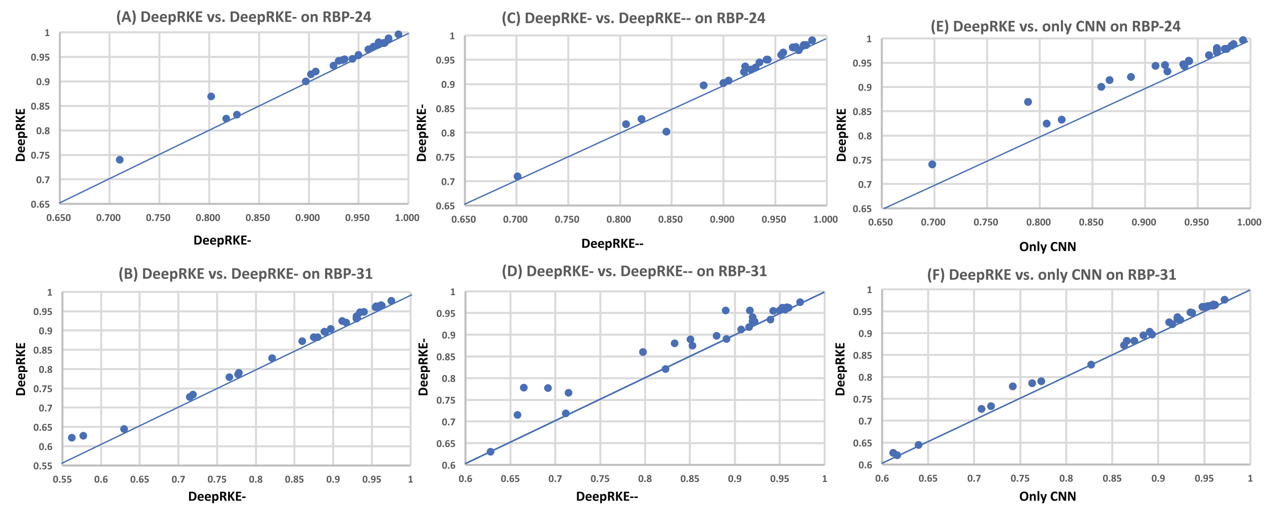

Fig. 3 Performance comparison of the models with or without distributed representation of sequences and secondary structural profiles. The performance was evaluated in terms of AUROC on RBP-24 and RBP-31 dataset. DeepRKE is our proposed model, DeepRKE- model is without RNA secondary structure, and DeepRKE- - is without RNA secondary structure and distributed representation of sequence, using one-hot encoding instead. **a**-**b** Performance comparison between DeepRKE and DeepRKE- on two datasets. **c**-**d** Performance comparison between DeepRKE- and DeepRKE- - on two datasets. **e**-**f** Performance comparison between models with only CNN larger and CNN+BiLSTM layer on two datasets

by one-hot encoding) and took one-hot encoding of RNA sequence as the input of deep learning framework. For convenience, we referred to this model as DeepRKE- -, indicating that both RNA secondary structure and distributed representations of RNA sequence are excluded when constructing the input.

Figure 3c-d shows the performance comparison between DeepRKE- and DeepRKE- - on dataset RBP-24 and RBP-31. We can see that the performance of DeepRKE- is better than DeepRKE- - on RBP-24 dataset, except for the CAPRIN1 protein. Specifically, DeepRKE- achieves an average AUC of 0.863, which is better than 0.841 of DeepRKE- - on RBP-31 dataset. DeepRKE- performs significantly better than DeepRKE- - on some proteins, e.g. Ago/EIF, Ago2-2, eIF4AIII-1, hnRNPL-2. The results suggest that the distributed representations of RNA sequence can significantly improve the performance for identifying binding site of RNA-binding proteins.

BiLSTM helps to improve prediction accuracy

In our proposed learning framework, BiLSTM is used to identify long-term dependent information of the extracted sequence and secondary structure feature [38]. For further evaluation of the performance enhancement by BiLSTM, we compared DeepRKE with a variant model using only CNN models that removes the BiLSTM layer. For the sake of fairness, we used the same parameters and architecture for CNN layer, including filter size, kernel size, learning rate and maxpool1d size. The results are shown in Fig. 3e-f, where DeepRKE achieves better performance than the CNN-only model on the two datasets. In particular, DeepRKE achieves an average AUC of 0.934 on RBP-24 dataset, which is better than that of 0.916 achieved by the CNN-only variant. Similarly, DeepRKE achieves an average AUC of 0.873 on RBP-31 dataset, which is still better than an average AUC of 0.863 of the CNN-only variant. The results suggest BiLSTM can effectively improve the performance of predicting protein-RNA binding sites.

Discussion

In this paper, we demonstrated that distributed representation can significantly improve the predictive ability in modeling protein-RNA binding. In fact, we can also obtain the distributed representations by using UTR sequences, and thus our method is not limited to the application of RBP-24 and RBP-31 benchmark datasets.

With the incorporation of biLTSM layer and secondary structure information, our method outperforms all counterpart methods in predicting protein-RNA binding sites. However, we can make advantage of RNA tertiary structure or region type to further improve the performance, similar to deepen-rbp [19], iDeep [41]. Compared to the CNN-only model, we also confirmed the advantage of recurrent neural networks in capturing high-order interdependence of sequences and secondary structures. In addition, we can also take the correlation between RBPs into consideration in the construction of our model, as done in protein-lncRNA interactions prediction [42] and protein-protein interactions prediction [43].

Note that only a small fraction of RNA sequences have been detected by CLIP-seq assays, the vast majority of sequences are not found so far. Therefore, the number of negative samples is much higher than that of positive samples in the real world. However, we built balanced training sets in this study, for that imbalanced datasets often lead to biased machine learning models with preference on overwhelming class (e.g. most of sequences are classified to negative samples), which makes trained models useless. More verified RBP binding sites are expected to improve the predictive power of computational models.

Although DeepRKE achieves state-of-the-art performance, it can not identify the binding motifs directly from learned convolve filters. We plan to extend DeepRKE to identify the binding motifs in our future work. For example, we can use the DeepRKE model to assign binding potential scores to all sequences of interest, and then select high-confidence candidates to extract the binding motifs.

Conclusion

In this paper, we present a novel deep neural network model, DeepRKE, to predict RBP binding sites. DeepRKE combines the primary RNA sequence and secondary structure into a unified learning framework. The novelty of DeepRKE lies in that we use the skip-gram model in word2vec to learn the distributed representation of RNA sequences and RNA structure. Also, we introduced the BiLSTM layer into the deep learning model to capture the high-order interdependence of sequences and secondary structures. We evaluated DeepRKE on two RBP binding benchmark sets derived from the CLIP-seq, and the results demonstrated that DeepRKE achieves better AUCs than other competitive methods. Our results suggest that distributed representation of k-mers sequence helps to improve the prediction performance for identifying the binding sites of RNA-binding protein. The BiLSTM layer also contributes significantly to the enhancement of predictive ability.

Methods

Datasets

Two large-scale datasets derived from human CLIP-seq: RBP-24 and RBP-31 are used as the benchmark datasets. The detail of the datasets are as below:

1) RBP-24 dataset is used by GraphProt [14] as the training and test set. It consists of 24 experiments covering 21 RBPs, and the RNA sequences are

variable length ranging from 150 to 375. For each experiment, the positive sites are downloaded from do-RiNA [40], the negative sites are created by shuffling the coordinates of binding sites within all genes with at least one binding site using bedtools shuffle [44]. We further randomly select a third of the original training data as validation set, and the remaining two-thirds as training set, and the independent testing set is the same test set used in GraphProt. The numbers of training positives and negatives of each experiment are listed in Table 2.

2) RBP-31 dataset includes training and test samples with fixed-length RNA sequences of 101 nucleotides collected in iONMF [15], which can be downloaded from https://github.com/mstrazar/ionmf. In this dataset, the CLIP-seq data consists of 19 proteins with 31 experiments, and their annotations are based on human assembly hg19. As described in iONMF, each nucleotide within clusters of interaction sites derived from CLIP-seq is considered as binding sites. To reduce redundancy, the positive binding sites with the highest cDNA count and without consecutive sites on genome are further randomly selected. Finally, among those sites with less than 15 nucleotides apart, one site with the highest cDNA counts was selected as the positive sample. The negative sites were sampled from genes that were not identified as an interacting partner in any of 31 experiments. As a result, 4,000 cross-linked sites are used for training, 1,000 samples for validation, and other 1,000 samples for independent test.

In addition, as DeepRKE requires RNA secondary structural sequence as input, we fed RNA sequence into RNAShapes [39] to obtain the dot parenthesis diagram, which is subsequently used as the input of EDeN to obtain the RNA secondary structure sequence. The RNAshapes have six generic shapes: stems (S), multiloops (M), hairpins (H), internal loops (I), dangling end (T) and dangling start (F).

Distributed representation of k-mer sequences

Word2Vec is a model proposed for learning semantic knowledge from a large number of textual corpora in an unsupervised manner, which is widely used in natural language processing. We extend the usage of word2vec to obtain the distribution representation of k-mer sequences. Based on the theory of distribution hypothesis, distributed representation is a strategy to obtain the semantic representation of words by using the symbiotic matrix. Word embedding is a multi-dimensional vector of real value which is mapped from the vocabulary words or phrases. It can capture the potential relationship between context and the target word. Generally, Word2Vec provides two architectural options: CBOW and skip-gram. CBOW can predict the current word based on the surrounding context, while the skip-gram uses the current word to predict the surrounding context. In this paper, we use skip-gram to learn the distributed representation of k-mers.

Given word sequence $s_1, s_2, s_3 .....s_k$, skip-gram learns the word representations using the co-occurrence information of words within a context window. It maximizes the following objective function:

$$\frac{1}{k}\sum_{i=1}^{k}\sum_{-m\leq j\leq m, j\neq 0} logp\left(S_{t+j} \mid S_j\right) \tag{1}$$

where m is the context window size, and the conditional probability p is defined as follows:

$$p\left(S_o \mid S_c\right) = \frac{exp\left(v'^{T}_{s_o} v_{s_c}\right)}{\sum_{s=1}^{V} exp\left(v'^{T}_{s} v_{s_c}\right)} \tag{2}$$

where V is the size of vocabulary, v_{s_c} is word vector of the center word, v_s and $\mathrm{v}_s^{'}$ is the input and output vector representation of word s, respectively.

Because of the computational infeasibility, the $logp(S_0 \mid S_i)$ is approximated using negative sampling:

$$log\sigma\left(v'^{T}_{s_o} v_{s_c}\right) + \sum_{i=1}^{k} E_{s_i \sim P_a(s)}\left[log\sigma\left(-v'^{T}_{s_i} v_{s_c}\right)\right] \tag{3}$$

where $\sigma = 1/(1 + exp(-x))$.

Recently, BioVec [34], seq2vec [33], dna2vec [32] and Gene2Vec [35] have also successively applied Word2Vec to encode biological sequences, and so did we. Specifically, we consider each k-mer as a word, each sequence as a sentence, all dataset sequences as corpus, as a result, we can learn the distributed representation of k-mers by using skip-gram algorithm. We split all sequences into 3-mer form. For example, the sequence AUUGC has 5 bp and its secondary structure is FHSIH, we can convert it to AUU,UUG,UGC,FHS,HSI,SIH. Based on the 3-mers derived from all training sequences, we trained the skip-gram algorithm using negative sampling, to predict the context of the targeted k-mer. Finally, we obtained the distributed representation for 4^3=64 (sequences) and 5^3=125 (structures) with 3-mers.

Learning framework of DeepRKE

Convolutional neural network: In our method, we employ three CNNs as feature extractors [45], which take as input the distribution representation of k-mer derived from RNA sequences and structures. The convolutional layer learns the weight parameters of the convolution filters, and outputs the matrix inner product between the input matrix and filters. After convolution, a rectified linear unit (ReLU) is applied to sparsify the output of the convolution layer and keep only positive matches to avoid the vanishing gradient problem [46]. Finally, a max pooling operation is used to reduce the dimensionality and yield invariance to small sequence shifts by pooling adjacent positions within a small window. DeepRKE includes three CNN modules, two of which are used to extract the features of RNA sequences and structures, respectively, and the third is designed to extract the high-order features between sequences and structures. The convolution kernel size of each CNN module is 8,16,32, and maxpooling size is 2.

Long Short Term Memory networks: In order to further detect long-term interdependence of sequence and secondary structure, we introduced a bidirectional long and short-term memory network (BiLSTM) [47] into our model. Compared to traditional recurrent neural networks (RNNs), LSTM has advantages in addressing the gradient vanishing or exploding, as well as capturing long-term dependency. Especially, BiLSTM exploits the contextual information on both sides.

As LSTM iterates over each element of input, it firstly determines what information the forgetting gate layer should exclude based on previous input. The input gate layer is then used to determine what information should be stored for the next layer and update the current state value. Finally, the output gate layer determines which parts of the state value should be output. Taking a sequence $\{x\}_{t=1}^{T}$ as input, the LSTM have the hidden states $\{h\}_{t=1}^{T}$, cell states $\{C\}_{t=1}^{T}$, and it outputs a sequence $\{o\}_{t=1}^{T}$. The above steps can be formulated as follows:

$$\begin{aligned} f_t &= \sigma\left(W_f x_t + U_f h_{t-1} + b_f\right), \\ i_t &= \sigma\left(W_i x_t + U_i h_{t-1} + b_i\right), \\ c_t &= f_t \odot c_{t-1} + i_t \odot tanh\left(W_c x_t + U_c h_{t-1} + b_c\right), \\ o_t &= \sigma\left(W_o x_t + U_o h_{t-1} + b_o\right), \\ h_t &= o_t \odot tanh(c_t) \end{aligned} \tag{4}$$

where $\odot$ denotes element-wise multiplication, the σ is the Logistic Sigmoid function and tanh is the tanh function to force the values to be between -1 and 1. W_f, W_i, W_o, U_f , U_i and U_o are the weights and b_f , b_i, b_c and b_o are the bias.

In DeepRKE, a bidirectional LSTM (BiLSTM) is used to scan from left to right and from right to left, concatenating the output in each direction as a final output for subsequent classification. The number of neurons in this layer was set to 32, thereby the output size is 64.

Supplementary Information

The online version contains supplementary material available at https://doi.org/10.1186/s12864-020-07239-w.

Additional file 1: The file includes the performance measures of competitive methods on the RBP-24 dataset. The AUC values of using DeepRKE- and without RNA secondary structure sequence, as well as the AUC values using DeepRKE- without RNA secondary structure and distributed representation of sequences.

Additional file 2: The file includes the performance measures of competitive methods on the RBP-31 dataset. The AUCs of using DeepRKE- and without RNA secondary structure sequence, as well as the AUC values of using DeepRKE- without RNA secondary structure and distributed representation of sequences.

Abbreviations

CLIP: UV-crosslinked immunoprecipatation; RBP: RNA binding proteins; PWM: Position weight matrix; GO: Gene ontology; CNN: Convolutional neural networks; LSTM: Long-term short-term memory networks; BiLSTM: Bidirectional long-term short-term memory networks; DBN: Deep belif neural network

Acknowledgements

Not applicable.

About this supplement

This article has been published as part of *BMC Genomics Volume 21 Supplement 13, 2020: Selected articles from the 2019 IEEE International Conference on Bioinformatics and Biomedicine (IEEE BIBM 2019): genomics (part 1)*. The full contents of the supplement are available online at https://bmcgenomics.biomedcentral.com/articles/supplements/volume-21-supplement-13.

Authors' contributions

LD and YZL proposed the learning framework. YZL implemented the model and performed the experiments. LD and YZL and wrote the paper. YCS and WHZ Participate in the preprocessing data module. YZL and LD designed the study and analyzed the obtained results. CY and HL supervised this work and improved the manuscript. All author(s) read and approved the final manuscript.

Funding

This work was supported by National Natural Science Foundation of China [grants No. 61972422, No. 61672113 and No. 61672541].Publication costs are funded by the No.61972422 grant from National Natural Science Foundation of China. The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

Availability of data and materials

The source code and data sets of DeepRKE are freely available at https://github.com/youzhiliu/DeepRKE/.

Ethics approval and consent to participate

Not applicable.

Consent for publication

Not applicable.

Competing interests

The authors declare that they have no competing interests.

Author details

[1]School of Computer Science and Engineering, Central South University, 410075, Changsha, China. [2]Aliyun School of Big Data, Changzhou University, 213164, Changzhou, China. [3]Department of Obstetrics, The Affiliated Changzhou No.2 People's Hospital of Nanjing Medical University, Changzhou, China.

Published: 17 December 2020

References

1. Keene JD. Rna regulons: coordination of post-transcriptional events. Nat Rev Genet. 2007;8(7):533.

2. Ray D, Kazan H, Cook KB, Weirauch MT, Najafabadi HS, Li X, Gueroussov S, Albu M, Zheng H, Yang A, et al. A compendium of rna-binding motifs for decoding gene regulation. Nature. 2013;499(7457):172.
3. de Silanes IL, Zhan M, Lal A, Yang X, Gorospe M. Identification of a target rna motif for RNA-binding protein hur. Proc Natl Acad Sci. 2004;101(9):2987–92.
4. Zhang W, Wagner B, Ehrenman K, Schaefer A, DeMaria C, Crater D, DeHaven K, Long L, Brewer G. Purification, characterization, and cdna cloning of an au-rich element rna-binding protein, auf1. Mol Cell Biol. 1993;13(12):7652–65.
5. Tiedje C, Diaz-Muñoz MD, Trulley P, Ahlfors H, Laaß K, Blackshear PJ, Turner M, Gaestel M. The rna-binding protein ttp is a global post-transcriptional regulator of feedback control in inflammation. Nucleic Acids Res. 2016;44(15):7418–40.
6. Vanderweyde T, Apicco DJ, Youmans-Kidder K, Ash PE, Cook C, da Rocha EL, Jansen-West K, Frame AA, Citro A, Leszyk JD, et al. Interaction of tau with the rna-binding protein tia1 regulates tau pathophysiology and toxicity. Cell Rep. 2016;15(7):1455–66.
7. Mukhopadhyay D, Houchen CW, Kennedy S, Dieckgraefe BK, Anant S. Coupled mrna stabilization and translational silencing of cyclooxygenase-2 by a novel rna binding protein, cugbp2. Mol Cell. 2003;11(1):113–26.
8. Licatalosi DD, Mele A, Fak JJ, Ule J, Kayikci M, Chi SW, Clark TA, Schweitzer AC, Blume JE, Wang X, et al. Hits-clip yields genome-wide insights into brain alternative rna processing. Nature. 2008;456(7221): 464–9.
9. Ray D, Kazan H, Chan ET, Castillo LP, Chaudhry S, Talukder S, Blencowe BJ, Morris Q, Hughes TR. Rapid and systematic analysis of the rna recognition specificities of RNA-binding proteins. Nat Biotechnol. 2009;27(7):667.
10. Van Nostrand EL, Pratt GA, Shishkin AA, Gelboin-Burkhart C, Fang MY, Sundararaman B, Blue SM, Nguyen TB, Surka C, Elkins K, et al. Robust transcriptome-wide discovery of rna-binding protein binding sites with enhanced clip (eclip). Nat Methods. 2016;13(6):508.
11. Hafner M, Landthaler M, Burger L, Khorshid M, Hausser J, Berninger P, Rothballer A, Ascano Jr M, Jungkamp A-C, Munschauer M, et al. Transcriptome-wide identification of rna-binding protein and microrna target sites by par-clip. Cell. 2010;141(1):129–41.
12. Kazan H, Ray D, Chan ET, Hughes TR, Morris Q. Rnacontext: a new method for learning the sequence and structure binding preferences of rna-binding proteins. PLoS Comput Biol. 2010;6(7):1000832.
13. Orenstein Y, Wang Y, Berger B. Rck: accurate and efficient inference of sequence-and structure-based protein–RNA binding models from rnacompete data. Bioinformatics. 2016;32(12):351–9.
14. Maticzka D, Lange SJ, Costa F, Backofen R. Graphprot: modeling binding preferences of RNA-binding proteins. Genome Biol. 2014;15(1):17.
15. Stražar M, Žitnik M, Zupan B, Ule J, Curk T. Orthogonal matrix factorization enables integrative analysis of multiple rna binding proteins. Bioinformatics. 2016;32(10):1527–35.
16. Livi CM, Blanzieri E. Protein-specific prediction of mrna binding using rna sequences, binding motifs and predicted secondary structures. BMC Bioinforma. 2014;15(1):123.
17. Corrado G, Tebaldi T, Costa F, Frasconi P, Passerini A. Rnacommender: genome-wide recommendation of RNA–protein interactions. Bioinformatics. 2016;32(23):3627–34.
18. Alipanahi B, Delong A, Weirauch MT, Frey BJ. Predicting the sequence specificities of dna-and rna-binding proteins by deep learning. Nat Biotechnol. 2015;33(8):831–8.
19. Zhang S, Zhou J, Hu H, Gong H, Chen L, Cheng C, Zeng J. A deep learning framework for modeling structural features of RNA-binding protein targets. Nucleic Acids Res. 2015;44(4):32.
20. Quang D, Xie X. Danq: a hybrid convolutional and recurrent deep neural network for quantifying the function of dna sequences. Nucleic Acids Res. 2016;44(11):107.
21. Pan X, Rijnbeek P, Yan J, Shen H-B. Prediction of rna-protein sequence and structure binding preferences using deep convolutional and recurrent neural networks. BMC Genomics. 2018;19(1):511.
22. Pan X, Shen H-B. Learning distributed representations of rna sequences and its application for predicting rna-protein binding sites with a convolutional neural network. Neurocomputing. 2018;305:51–58.
23. Budach S, Marsico A. pysster: classification of biological sequences by learning sequence and structure motifs with convolutional neural networks. Bioinformatics. 2018;34(17):3035–7.
24. Ben-Bassat I, Chor B, Orenstein Y. A deep neural network approach for learning intrinsic protein-rna binding preferences. Bioinformatics. 2018;34(17):638–46.
25. LeCun Y, Bengio Y, Hinton G. Deep learning. Nature. 2015;521(7553): 436–44.
26. He K, Zhang X, Ren S, Sun J. Deep residual learning for image recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Las Vegas: IEEE; 2016. p. 770–8.
27. Zhang X, Zhao J, LeCun Y. Character-level convolutional networks for text classification. In: Advances in Neural Information Processing Systems. Montreal: MIT Press; 2015. p. 649–57.
28. Hannun A, Case C, Casper J, Catanzaro B, Diamos G, Elsen E, Prenger R, Satheesh S, Sengupta S, Coates A, et al. Deep speech: Scaling up end-to-end speech recognition. arXiv preprint arXiv:1412.5567. 2014.
29. Mamoshina P, Vieira A, Putin E, Zhavoronkov A. Applications of deep learning in biomedicine. Mol Pharm. 2016;13(5):1445–54.
30. Lawrence S, Giles CL, Tsoi AC, Back AD. Face recognition: A convolutional neural-network approach. IEEE Trans Neural Netw. 1997;8(1):98–113.
31. Lee H, Grosse R, Ranganath R, Ng AY. Convolutional deep belief networks for scalable unsupervised learning of hierarchical representations. In: Proceedings of the 26th Annual International Conference on Machine Learning. Montreal: ACM; 2009. p. 609–16.
32. Ng P. dna2vec: Consistent vector representations of variable-length k-mers. arXiv preprint arXiv:1701.06279. 2017.
33. Kimothi D, Soni A, Biyani P, Hogan JM. Distributed representations for biological sequence analysis. arXiv preprint arXiv:1608.05949. 2016.
34. Asgari E, Mofrad MR. Continuous distributed representation of biological sequences for deep proteomics and genomics. PloS ONE. 2015;10(11): 0141287.
35. Zou Q, Xing P, Wei L, Liu B. Gene2vec: gene subsequence embedding for prediction of mammalian n6-methyladenosine sites from mrna. RNA. 2019;25(2):205–18.
36. Goldberg Y, Levy O. word2vec explained: deriving mikolov et al.'s negative-sampling word-embedding method. arXiv preprint arXiv:1402.3722. 2014.
37. Guthrie D, Allison B, Liu W, Guthrie L, Wilks Y. A closer look at skip-gram modelling. In: LREC. Genoa: ELRA; 2006. p. 1222–5.
38. Huang Z, Xu W, Yu K. Bidirectional lstm-crf models for sequence tagging. arXiv preprint arXiv:1508.01991. 2015.
39. Steffen P, Voß B, Rehmsmeier M, Reeder J, Giegerich R. Rnashapes: an integrated rna analysis package based on abstract shapes. Bioinformatics. 2005;22(4):500–3.
40. Anders G, Mackowiak SD, Jens M, Maaskola J, Kuntzagk A, Rajewsky N, Landthaler M, Dieterich C. dorina: a database of rna interactions in post-transcriptional regulation. Nucleic Acids Res. 2011;40(D1):180–6.
41. Pan X, Shen H-B. Rna-protein binding motifs mining with a new hybrid deep learning based cross-domain knowledge integration approach. BMC Bioinforma. 2017;18(1):136.
42. Zhang J, Zhang Z, Wang Z, Liu Y, Deng L. Ontological function annotation of long non-coding rnas through hierarchical multi-label classification. Bioinformatics. 2017;34(10):1750–7.
43. Deng L, Guan J, Dong Q, Zhou S. Prediction of protein-protein interaction sites using an ensemble method. BMC Bioinforma. 2009;10(1):426.
44. Quinlan AR, Hall IM. Bedtools: a flexible suite of utilities for comparing genomic features. Bioinformatics. 2010;26(6):841–2.
45. Sharif Razavian A, Azizpour H, Sullivan J, Carlsson S. Cnn features off-the-shelf: an astounding baseline for recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops. Columbus: IEEE; 2014. p. 806–13.
46. Ramachandran P, Zoph B, Le QV. Searching for activation functions. arXiv preprint arXiv:1710.05941. 2017.
47. Graves A, Schmidhuber J. Framewise phoneme classification with bidirectional lstm and other neural network architectures. Neural Netw. 2005;18(5-6):602–10.

Publisher's Note