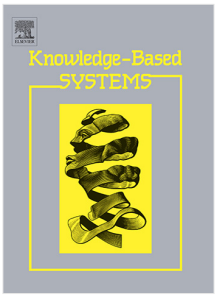
Journal Pre-proof



NeuroTIS: Enhancing the prediction of translation initiation sites in mRNA sequences via a hybrid dependency network and deep learning framework

Chao Wei, Junying Zhang, Xiguo Yuan, Zongzhen He, Guojun Liu, Jinhui Wu

PII: S0950-7051(20)30588-8
DOI: https://doi.org/10.1016/j.knosys.2020.106459
Reference: KNOSYS 106459

To appear in: *Knowledge-Based Systems*

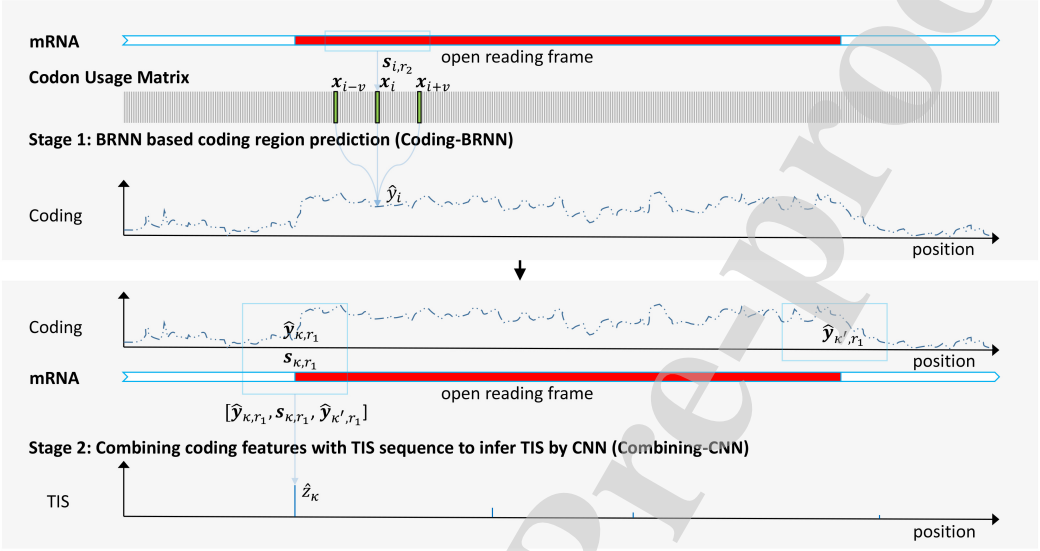
Received date : 5 March 2020
Revised date : 4 August 2020
Accepted date : 2 September 2020

Please cite this article as: C. Wei, J. Zhang, X. Yuan et al., NeuroTIS: Enhancing the prediction of translation initiation sites in mRNA sequences via a hybrid dependency network and deep learning framework, *Knowledge-Based Systems* (2020), doi: https://doi.org/10.1016/j.knosys.2020.106459.

Graphical Abstract

NeuroTIS: Enhancing the prediction of translation initiation sites in mRNA sequences via a hybrid dependency network and deep learning framework

Chao Wei,Junying Zhang,Xiguo Yuan,Zongzhen He,Guojun Liu,Jinhui Wu



Highlights

NeuroTIS: Enhancing the prediction of translation initiation sites in mRNA sequences via a hybrid dependency network and deep learning framework

Chao Wei,Junying Zhang,Xiguo Yuan,Zongzhen He,Guojun Liu,Jinhui Wu

- Exploiting label dependency among coding region helps prediction of coding region.
- Exploiting label dependency between coding region and TIS helps prediction of TIS.
- Incorporating explicit biological knowledge promotes power of CNN for TIS prediction.

NeuroTIS: Enhancing the prediction of translation initiation sites in mRNA sequences via a hybrid dependency network and deep learning framework⋆,⋆⋆

Chao Wei, Junying Zhang*, Xiguo Yuan, Zongzhen He, Guojun Liu and Jinhui Wu

School of Computer Science and Technology, Xidian University, Xi'an 710071, PR China

ARTICLE INFO

Keywords:
Deep learning
Bioinformatics
Translation initiation sites prediction
Dependency network
Label dependency

ABSTRACT

Translation initiation site prediction is crucial to understand the mechanisms of gene expression and regulation. Many computational approaches have been proposed and achieved acceptable prediction accuracy. Although recent Convolutional Neural Network-based method effectively learn consensus motifs and shows remarkable prediction performance, this method could not fully exploit coding features which have been proved significant to the identification of translation initiation sites. Indeed, coding features often exhibit higher-order distant interactions among nucleotides and learning this kind of feature from uncharacteristic mRNA sequences without any explicit biological knowledge is difficult. This situation gets worse when given no coding labels. In viewing of these shortcomings, we propose a novel method for translation initiation sites prediction in mRNA sequences based on a hybrid dependency network and deep learning framework (NeuroTIS) which explicitly model label dependencies among coding region, between coding region and translation initiation site. Meanwhile, a Bidirectional Recurrent Neural Network and a Convolutional Neural Network are employed for effective learning and inference. The experimental results show that the proposed framework yields an excellent prediction performance on two benchmark gene datasets, which significantly outperforms existing state-of-the-art methods.

1. Introduction

Translation initiation plays a significant role in mRNA translation and protein synthesis, the dysregulation of the initiation process can cause various human diseases, including cancers and metabolic disorders Sonenberg and Hinnebusch (2009); Barbosa, Peixeiro and Romão (2013); Zhang, Hu, Jiang, Zhang and Zeng (2017). On the other hand, the development of next-generation sequencing (NGS) technologies give rise to exponential increase of sequence data. Many efforts have been dedicated to the identification of genomic mutations by using NGS datasets Yuan, Zhang, Yang, Bai and Fan (2017); Yuan, Bai, Zhang, Yang, Duan, Li and Gao (2018); Xi, Li and Wang (2018, 2017) in the past few years, it is urgent to find reliable translation initiation sites (TISs) prediction methods for understanding the complex mechanisms of gene expression and regulation.

TISs are the positions in mRNA sequences to start constructing proteins. The graphical illustration of translation initiation is shown in Fig. 1. Identification of TISs from uncharacterized mRNA sequences is a challenging task. This is because (1) the conserved tri-nucleotide AUG is not sufficient to determine the true TISs due to presence of a large number of AUG in genes, which induces considerable false positives; (2) the TISs is not always conserved with tri-nucleotide AUG, a few exceptions are reported in eukaryotes Pedersen and Nielsen (1997); Hatzigeorgiou (2002); (3) unlike highly-conserved splicing signals, TISs are surrounded by relatively poorly conserved sequences and harder to predict Bernal, Crammer, Hatzigeorgiou and Pereira (2007); (4) the scanning mechanism of translation initiation is complex Pelletier and Sonenberg (1988) and the scanning model Kozak (1989) is not always applicable. For example, many mRNAs contain multiple open reading frames, including upstream open reading frames (uORFs), which generally repress translation of the downstream ORF Hinnebusch, Ivanov and Sonenberg (2016); Boersma, Khuperkar, Verhagen, Sonneveld, Grimm, Lavis and Tanenbaum (2019); Khuperkar, Hoek, Sonneveld, Verhagen, Boersma and Tanenbaum (2020). Fortunately, there are biased codon usages around TISs, which make it possible to identify TISs by computational methods.

*Corresponding author.
jyzhang@mail.xidian.edu.cn (J. Zhang)
ORCID(s): 0000-0002-7511-0277 (J. Zhang)

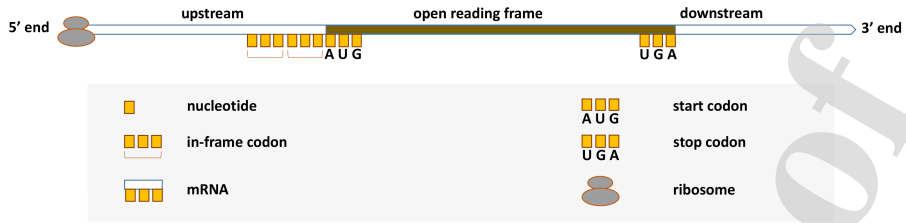


Figure 1: The initiation of translation. The ribosome scans the mRNA until it reads an AUG codon that has appropriate context.

Many previous efforts have been exerted on the prediction of TISs including Artificial Neural Networks (ANN) Pedersen and Nielsen (1997); Rajapakse and Ho (2005); Zhang et al. (2017); Zuallaert, Kim, Soete, Saeys and Neve (2018b), Support Vector Machine (SVM) Zien, Rätsch, Mika, Schölkopf, Lengauer and Müller (2000); Li and Jiang (2004); Chen, Feng, Deng, Lin and Chou (2014), Linear Discriminant Analysis (LDA) Salamov (1998) and the Gaussian Model Li, Leong and Zhang (2005). In one of works, Pedersen and Nielsen (1997) directly feed DNA sequences into an ANN to predict TISs in mRNA sequences, based on a window of 200 bp centered on the AUG. It is discovered that position -3 is crucial in the identification of TISs, which corroborates with the other studies cited. They also discover that ATGs that are in-frame to TISs are more likely to be predicted incorrectly as TISs regardless of whether they are upstream or downstream of the AUG. The work Salamov (1998) develops a system called ATGpr to identify TISs with LDA, which exploits six effective characteristics around ATG including the positional triplet weight matrix and the ORF hexanucleotide characteristics. The next version of ATGpr called ATGpr_sim is developed in Nishikawa, Ota and Isogai, which uses both statistical information and similarities with other known proteins to obtain higher prediction performance in cDNA sequences. Zien et al. (2000) studies SVM using different kinds of kernel functions for TISs prediction. They believe that carefully designing kernel functions are useful for achieving higher prediction accuracy. Recently, deep learning has been effectively applied to biological data, such as functional sites recognition Solovyev and Umarov (2016); Min, Zeng, Chen, Chen, Chen and Jiang (2017); Yi, Liu, Macleod and Liu (2017); Du, Yao, Diao, Zhu, Zhang and Li (2018); Zuallaert, Godin, Kim, Soete, Saeys and De Neve (2018a); Zhang et al. (2017); Zuallaert et al. (2018b), the prediction of DNA- and RNA-binding proteins Alipanahi, Delong, Weirauch and Frey (2015) and the unwinding capability of mRNA structure *in vivo* Yu, Meng, Mao, Zhang, Sun and Tao (2019). Zuallaert et al. (2018b) develop a method called TISRover, which directly learn biological knowledge (e.g., the Kozak consensus sequence Kozak (1983), the reading frame characteristics and the presence of donor splice site patterns) from DNA sequences with a Convolutional Neural Network (CNN).

Although the above-mentioned methods for TISs prediction achieve acceptable prediction accuracy, they still have some shortcomings. Indeed, the most relevant feature for TISs prediction is the transition from a non-coding region to a coding region in the first reading frame Saeys (2004); Li, Liu, Wong and Yap (2004). However, these methods might not fully exploit this feature. First, all the above-mentioned methods are usually trained by a set of data which contain labeled true and false TISs, but they ignore the co-existence of labels of coding region. The prediction result might be suboptimal due to the ignorance of this additional label information. Second, as pointed out by Rajapakse and Ho (2005); Li et al. (2004), it is not easy for neural networks to learn high-order correlations from extremely low level inputs, e.g., a string of nucleotides. They Rajapakse and Ho (2005) alleviate this problem by a Markov encoding technique which explicitly incorporate known biological characteristics in genomic sequences to encode the inputs to neural network and significantly improve the prediction performance.

Based on the aforementioned analysis, we explore how to enhance the prediction of TISs in this article. We propose a novel method for translation initiation sites prediction in mRNA sequences based on a hybrid dependency network Heckerman, Chickering, Meek, Rounthwaite and Kadie (2013) and deep learning Lecun, Bengio and Hinton (2015) framework (NeuroTIS) which *explicitly* model the label dependency among coding region, between TISs and coding region. Moreover, it allows for a simple learning phase and an approximate inference by utilizing a Bidirectional Recurrent Neural Networks (BRNN) and a CNN. Evaluated on two benchmark datasets, our method yields an excellent prediction performance, which significantly outperform existing state-of-the-art methods. There are three main contributions which may explain the excellent performance of our proposed framework:

- The NeuroTIS is a marriage between dependency network and deep learning, which could exploit the joint merits. The dependency network *explicitly* model label dependencies among coding region, between TISs and coding region, whereas deep learning effectively and automatically learn label dependencies, coding features and consensus motifs.
- We exploit label constraint among coding region in the first time and explore two BRNN architectures for coding region prediction, which significantly improve the prediction performance over traditional methods that utilise the codon usage statistic only.
- Compared with neural network-based methods, multiple explicit biological features, e.g., codon usage, the scanning model, are employed and significantly improve the performance of TISs prediction. Moreover, all features are fed into a CNN, which might facilitates feature interactions.

The source code and the dataset used in the paper are publicly available at: https://github.com/xdcwei/NeuroTIS/.

2. Related works

We here briefly review the most relevant works to ours. Indeed, the idea that prediction of coding region helps in predicting signal sites has emerged in some previous works. It is firstly introduced for splice site prediction in Brunak, Engelbrecht and Knudsen (1991), who proposes a method called NetGene which combines both local and global sequence information in neural networks. It is based on their observation of complementary relation that short exons tend to have strong consensus splice sites, while long exons allow for weaker splice sites. A new method based on a modular system of neural networks to identify eukaryotic gene structure is proposed in Hatzigeorgiou, Mache and Reczko (1996). The prediction task is divided into the detection of distinct signals and content based on different neural network architectures. The work Hatzigeorgiou (2002) develops a multi-step integrated method for TISs prediction by combining a consensus neural network sensitive to conserved motif and a coding neural network sensitive to the coding/noncoding potential around the start codon. In the work of Tzanis, Berberidis and Vlahavas (2007), the authors propose a modular method for the prediction of TISs, called MANTIS, with three main components: Consensus, Coding Region classification, and ATG Location. The components are conducted by three classifiers respectively and the prediction scores are fused into a decision classifier at the final stage of MANTIS. The enhanced version of this method called StackTIS, which adopts different learning procedure and training strategy is reported in Tzanis, Berberidis and Vlahavas (2012). All these works give us a strong intuition that predictions of protein coding region and TISs are two highly-correlated tasks which exploit two different kinds of features but exhibit correlations in label space, one task can help the other task be completed better.

Our work differs from the above-mentioned works in four aspects. First, we provide a natural probabilistic interpretation of label dependencies among coding region, between TISs and coding region in the first time. Second, unlike the above-mentioned works using ANN, a BRNN and a CNN are employed to extract coding features and consensus motifs, which achieves significant performance improvement. Third, existing methods usually extract coding features and consensus motifs by using multiple classifiers and combine each score for final prediction, while we integrate coding features with TISs sequence directly into CNN, which allows for automatic learning and facilitates feature interactions. Fourth, none of the above works exploit the coding features around stop codon. Indeed, for a TIS, the region around its first in-frame stop codon is transferred from coding to non-coding in the first reading frame. Hence, the coding features around stop codon is also relevant to a candidate TIS.

3. The NeuroTIS framework

In this section, definitions of problem, dependency network Heckerman et al. (2013) representation of NeuroTIS, the inference and learning phases are introduced. The graphical illustration of NeuroTIS is shown in Fig. 2 and 3.

3.1. Preliminaries

In what follows, $s = s_1 s_2 ... s_n$ is a mRNA sequence and $\boldsymbol{z} = z_1 z_2 ... z_n$ is the label sequence of $\boldsymbol{s}$, where $s_i \in \{A, C, T, G\}$ and $z_i \in \{1, 0\}$. Let $\boldsymbol{s}_{p,r}$ indicate a subsequence of $\boldsymbol{s}$ centered at position p with a fixed length window $2 \times r + 1$, and then the TISs prediction is equivalent to solve the following *maximum a posteriori* (MAP) estimation problem

$$z_\kappa^* = \arg\max_{z_\kappa} p(z_\kappa | \boldsymbol{s}_{\kappa, r_1}) \tag{1}$$

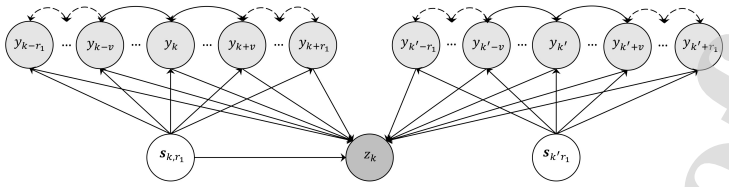


Figure 2: The dependency network representation of NeuroTIS in which nodes denote random variables and arcs denote probabilitistic dependencies between variables.

where r_1 denotes the half-length of TIS sequence, we assume that κ denotes the position of k-th tri-nucleotides ATG in the sequence $\boldsymbol{s}$, and z_κ denotes whether the position κ in $\boldsymbol{s}$ is true TIS ($z_\kappa = 1$) or not ($z_\kappa = 0$).

Almost all machine learning-based methods regard TISs prediction as an independent binary classification problem and learn the conditional probability Eq. (1) to infer TISs. However, they ignore potential dependency among labels. Indeed, there exists multiple kinds of label dependencies for TISs prediction in a mRNA sequence. First, a TIS is the start codon of open reading frame, where a non-coding region is transferred to a coding region in the first reading frame. Hence, there is label dependency between a TIS and local coding region around the TIS. Second, translation initiates at a TIS and terminates at the first in-frame stop codon of the TIS, and thus there is a label dependency between the TIS and local coding region around stop codon. Third, open reading frame is a long region, and hence there exists label dependency among the long coding region. Our intuition is to *explicitly* model these label dependencies via a natural dependency network representation, in which case we can better predict TIS and coding region simultaneously. Hence, following the probability representation in multi-label classification Read, Martino and Hollmen (2017); Guo and Gu (2011), we here consider the following MAP problem,

$$(z^*_\kappa, \boldsymbol{y}^*_{\kappa,r_1}, \boldsymbol{y}^*_{\kappa',r_1}) = \arg \max_{z_\kappa, \boldsymbol{y}_{\kappa,r_1}, \boldsymbol{y}_{\kappa',r_1}} p(z_\kappa, \boldsymbol{y}_{\kappa,r_1}, \boldsymbol{y}_{\kappa',r_1} | \boldsymbol{s}_{\kappa,r_1}, \boldsymbol{s}_{\kappa',r_1}) \tag{2}$$

where $\boldsymbol{y} = y_1 y_2 ... y_n$ is the label sequence of $\boldsymbol{s}$ and $y_i \in \{1, 0\}$ denotes whether the position i in $\boldsymbol{s}$ is coding ($y_i = 1$) or not ($y_i = 0$). κ' denotes the position of the first in-frame stop codon with respect to the k-th candidate TIS.

3.2. Dependency network representation

We adopt a dependency network to encode the potential label dependencies among coding region, between coding region and TIS. Dependency network is a type of cyclic directed graphical model Koller and Friedman (2009), where the directed edges encodes not ordered relationships like Bayesian network Pearl (1986) but directed dependencies among variables Guo and Gu (2011). Fig. 1 shows the dependency network representation of NeuroTIS. From this figure one can see that each label of coding region is directly dependent on its two neighboring coding labels and TIS sequence, whereas label of TIS is dependent on coding labels around TIS, coding labels around in-frame stop codon, and TIS sequence. By the chain rule of probability Schum (1994), the conditional joint probability of $(z_\kappa, \boldsymbol{y}_{\kappa,r_1}, \boldsymbol{y}_{\kappa',r_1})$ given the sequences $\boldsymbol{s}_{\kappa,r_1}$ and $\boldsymbol{s}_{\kappa',r_1}$ can be expressed as:

$$p(z_\kappa, \boldsymbol{y}_{\kappa,r_1}, \boldsymbol{y}_{\kappa',r_1} | \boldsymbol{s}_{\kappa,r_1}, s_{\kappa',r_1}) = p(z_\kappa | \boldsymbol{s}_{\kappa,r_1}, \boldsymbol{y}_{\kappa,r_1}, \boldsymbol{y}_{\kappa',r_1}) p(\boldsymbol{y}_{\kappa,r_1} | \boldsymbol{s}_{\kappa,r_1}) p(\boldsymbol{y}_{\kappa',r_1} | \boldsymbol{s}_{\kappa',r_1}) \tag{3}$$

From Eq. (3), one can find that the presence of coding region affect the probability of TIS. Note that here we make a simple assumption of unidirectional dependency between TIS and coding region for efficient learning and inference. However, the conditional joint probability distributions $p(\boldsymbol{y}_{\kappa,r_1} | \boldsymbol{s}_{\kappa,r_1})$ and $p(\boldsymbol{y}_{\kappa',r_1} | \boldsymbol{s}_{\kappa',r_1})$ are more complicated to estimate for bidirectional dependency among coding labels, we here adopt a Gibbs sampling Geman (1984) that approximate them with a product of univariate conditional distributions as follows:

$$p(\boldsymbol{y}_{\kappa,r_1} | \boldsymbol{s}_{\kappa,r_1}) \approx \prod_{i=-r_1}^{r_1} p(y_{\kappa+i} | \boldsymbol{s}_{\kappa,r_1}, y_{\kappa+i-v}, y_{\kappa+i+v}) \tag{4}$$

$$p(\boldsymbol{y}_{\kappa',r_1} | \boldsymbol{s}_{\kappa',r_1}) \approx \prod_{i=-r_1}^{r_1} p(y_{\kappa'+i} | \boldsymbol{s}_{\kappa',r_1}, y_{\kappa'+i-v}, y_{\kappa'+i+v}) \tag{5}$$

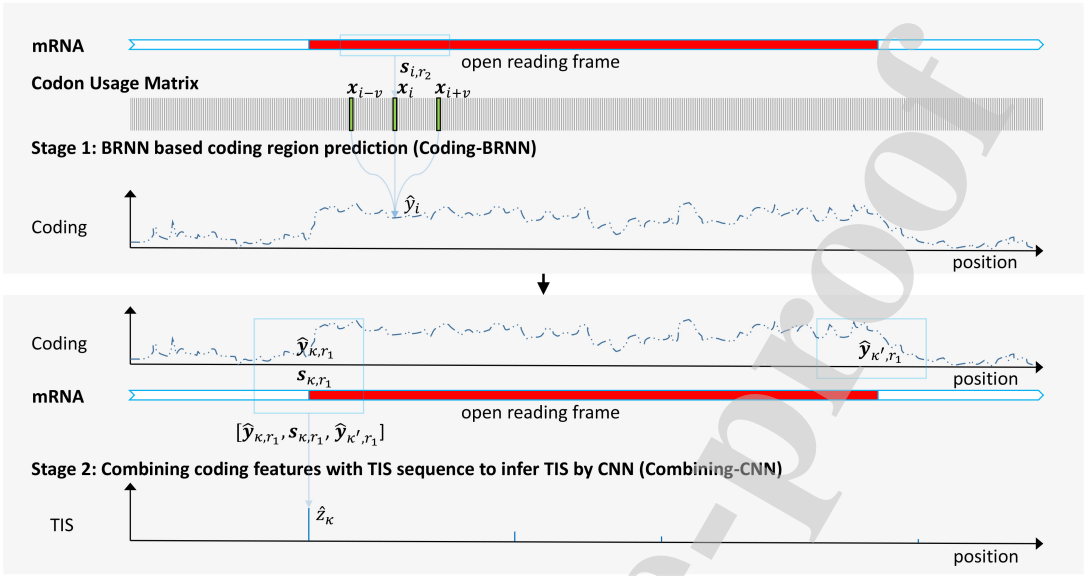


Figure 3: The greedy inference procedure of NeuroTIS. It consists of two stages: Coding-BRNN and Combining-CNN. At the first stage, Coding-BRNN infers each coding score for a sliding window, and at the second stage, Combining-CNN combines two kinds of coding scores with TIS sequence to infer TIS.

where v is a step interval that defines how far that two nodes correlate and its detailed description is given in subsection 3.4. The key to the Gibbs sampling is that one can approximate the joint distribution by sampling one variable at a time with the other variables fixed. This property makes Gibbs sampling a very simple inference algorithm and particularly well-adapted to dependency network where there exists cyclic dependencies.

3.3. Inference

The other basic problem of dependency network is how to infer variables $(z_\kappa, \boldsymbol{y}_{\kappa,r_1}, \boldsymbol{y}_{\kappa',r_1})$ given the sequences $\boldsymbol{s}_{\kappa,r_1}$ and $\boldsymbol{s}_{\kappa',r_1}$. From the Eq. (2)-(5), the problem of jointly infer TIS and coding region is reduced to solve a type of *maximum a posteriori* (MAP) problem

$$(z^*_\kappa, \boldsymbol{y}^*_{\kappa,r_1}, \boldsymbol{y}^*_{\kappa',r_1}) = \arg \max_{z_\kappa, \boldsymbol{y}_{\kappa,r_1}, \boldsymbol{y}_{\kappa',r_1}} p(z_\kappa|\boldsymbol{s}_{\kappa,r_1}, \boldsymbol{y}_{\kappa,r_1}, \boldsymbol{y}_{\kappa',r_1}) \prod_{i=-r_1}^{r_1} p(y_{\kappa+i}|\boldsymbol{s}_{\kappa,r_1}, y_{\kappa+i-v}, y_{\kappa+i+v}) p(y_{\kappa'+i}|\boldsymbol{s}_{\kappa',r_1}, y_{\kappa'+i-v}, y_{\kappa'+i+v}) \tag{6}$$

Note that this problem is NP-hard and exact inference is intractable, which induces exponential complexity, in searching all of the $2^{(4\times r_1+3)}$ paths. Hence, we make a simple and efficient greedy inference to approximate Eq. (6) as follows

$$\hat{y}_{\kappa,r_1} = \arg \max_{\boldsymbol{y}_{\kappa,r_1}} \prod_{i=-r_1}^{r_1} p(y_{\kappa+i}|\boldsymbol{s}_{\kappa,r_1}, y_{\kappa+i-v}, y_{\kappa+i+v}) \tag{7}$$

$$\hat{\boldsymbol{y}}_{\kappa',r_1} = \arg \max_{\boldsymbol{y}_{\kappa',r_1}} \prod_{i=-r_1}^{r_1} p(y_{\kappa'+i}|\boldsymbol{s}_{\kappa',r_1}, y_{\kappa'+i-v}, y_{\kappa'+i+v}) \tag{8}$$

$$\hat{z}_\kappa = \arg \max_{z_\kappa} p(z_\kappa|\boldsymbol{s}_{\kappa,r_1}, \hat{\boldsymbol{y}}_{\kappa,r_1}, \hat{\boldsymbol{y}}_{\kappa',r_1}) \tag{9}$$

Using Eq. (7)-(9), NeuroTIS can efficiently infer variables $(z_\kappa, \boldsymbol{y}_{\kappa,r_1}, \boldsymbol{y}_{\kappa',r_1})$. The inference procedure can be simply performed in two stages: one stage to predict coding region $\hat{\boldsymbol{y}}_{\kappa,r_1}$ and $\hat{\boldsymbol{y}}_{\kappa',r_1}$, respectively, and the second stage to combine predicted coding scores at the first stage with TIS sequence $[\hat{\boldsymbol{y}}_{\kappa,r_1}, \boldsymbol{s}_{\kappa,r_1}, \hat{\boldsymbol{y}}_{\kappa',r_1}]$ to jointly infer the TIS

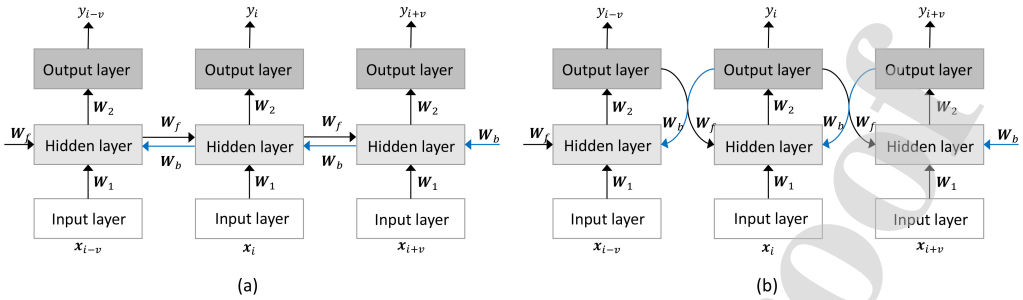


Figure 4: The two different BRNN architectures for coding region prediction in mRNA sequences. (a) Coding-BRNN1; (b) Coding-BRNN2. The main difference of Coding-BRNN2 from Coding-BRNN1 lies in that Coding-BRNN2 re-injects bidirectional predicted labels as input into the network.

$\hat{z}_\kappa$. Although Gibbs sampling could easily solve Eq. (7) and (8), the inference procedure must be performed iteratively. In practice, we employ a BRNN to estimate conditional probability distributions $p(y_{\kappa+i}|\boldsymbol{s}_{\kappa,r_1}, y_{\kappa+i-v}, y_{\kappa+i+v})$ and $p(y_{\kappa'+i}|\boldsymbol{s}_{\kappa',r_1}, y_{\kappa'+i-v}, y_{\kappa'+i+v})$. In contrast to Gibbs sampling, inference procedure of BRNN is performed only once when training process is finished. As for the conditional probability distribution $p(z_\kappa|\hat{\boldsymbol{y}}_{\kappa,r_1}, \boldsymbol{s}_{\kappa,r_1}, \hat{\boldsymbol{y}}_{\kappa',r_1})$, we estimate it by utilising a CNN. The details are described in the following sections.

3.4. Bidirectional Recurrent Neural Network for coding region prediction

3.4.1. Codon Usage Matrix

Many previous methods Brunak et al. (1991); Hatzigeorgiou et al. (1996) directly employ genomic sequence as input into a classifier for coding region prediction. There might be performance benefits in eukaryotic genes where exons are interrupted by introns, however, in mRNA or cDNA, it has been shown that preprocessing the data through a coding measure can significantly improve the performance of the ANN. Through a variety of existing coding measure methods the best results were obtained by applying the codon usage statistic which is calculated for a sliding window of transcript sequences Hatzigeorgiou (2002). Indeed, as claimed in Fickett and Tung (1992), there is a great deal of redundancy in current coding measures and effective coding measures are confined to some counting oligomers, e.g., coding usage, hexamer usage. Hence, we employ codon usage statistic as the features for coding region prediction in mRNA sequences. We adopt the same strategy as Hatzigeorgiou (2002); Tzanis et al. (2012) to generate training and test samples. A window of 84 nucleotides ($r_2 = 42$) is sliding over the mRNA sequences. The counting starts with the first nucleotide of the window, counting all non-overlapping codons, in which case $\boldsymbol{s}_{i,r_c}$ is converted into a 64 dimensional vector $\boldsymbol{x}_i$. For each mRNA sequence with a length of n nucleotides, a codon usage matrix with 64 rows and n columns is generated. The graphical illustration is shown in Fig. 3.

3.4.2. Bidirectional Recurrent Neural Network

Recurrent neural network (RNN) Goodfellow, Bengio and Courville (2016) is a class of artificial neural network dealing with time sequential data that exhibits correlations between time points. For tasks that involve sequential inputs, such as speech and language, it is often better to use RNN. RNN process an input sequence one element at a time point, maintaining in their hidden units a 'state vector' that implicitly contain information about the history of all the past elements of the sequence. Bidirectional Recurrent Neural Network (BRNN) Schuster and Paliwal (1997) extends the RNN and can be trained using all available input information in the past and future of a specific time point. This is implemented by two separate procedures that are responsible for the positive time direction (forward pass) and negative time direction (backward pass) respectively.

In our work, we are inspired by the idea that exploitation of the past and future input information will improve the prediction performance of time sequential data. We propose a BRNN-based method for coding region prediction (Coding-BRNN). The main difference of Coding-BRNN from Coding-ANN lies in that the former one exploit statistical dependency among mRNA sequences, while the latter one only consider local biological features, e.g., codon usage statistic. Indeed, for a sample $\boldsymbol{x}_i$ to be predicted, if its neighboring samples ($\boldsymbol{x}_{i-v}$ and $\boldsymbol{x}_{i+v}$) is coding (non-coding), there will be high probabilities that it is also coding (non-coding). We have a strong intuition that exploitation of this

statistical dependency can yield further improvement on coding region prediction.

There exists two simple types of RNNs according to the position of recurrent connection: the Elman Elman (1990) and Jordan Jordan (1986) RNN models. The former one performs the recurrent connection in the hidden layers, whereas in the Jordan RNN it connects the output layer to the hidden layer. The two networks are widely used in sequence labeling problems and often achieves comparable performance Mesnil, He, Deng and Bengio (2013); Dinarelli and Tellier (2016). In our opinion, both of the two models consider statistical dependency in time series, Elman network utilize the dependency in feature space while Jordan network in label space, it is hard to decide which model is superior, this depends on concrete applications—sometimes Elman RNN better Pham and Karaboga (1999), and sometimes Jordan RNN better Errattahi, El Hannani, Salmam and Ouahmane (2019). Hence, we here explore two types of network architectures for BRNN: Coding-BRNN1 and Coding-BRNN2. As shown in Fig. 4, they share a similar network architecture and the main difference of Coding-BRNN2 from Coding-BRNN1 lies in that Coding-BRNN2 *explicitly* re-inject predicted labels as input into the network. The forward and backward pass of Coding-BRNN1 and Coding-BRNN2 can be respectively formulated as:

$$I_{i-v} = \sigma_1(W_I(I_0 \oplus \boldsymbol{x}_{i-v})) \tag{10}$$

$$J_{i+v} = \sigma_1(W_J(U_0 \oplus \boldsymbol{x}_{i+v})) \tag{11}$$

and

$$I_{i-v} = \sigma_2(W_2(\sigma_1(W_I(I_0 \oplus \boldsymbol{x}_{i-v})))) \tag{12}$$

$$J_{i+v} = \sigma_2(W_2(\sigma_1(W_J(J_0 \oplus \boldsymbol{x}_{i+v})))) \tag{13}$$

then for a sample $\boldsymbol{x}_i$, the feedbacks I_{i-v} and J_{i+v} from its neighboring samples are both concatenated for the final prediction $\hat{y}_i$:

$$\hat{y}_i = \sigma_2(W_2(\sigma_1(W_y(I_{i-v} \oplus \boldsymbol{x}_i \oplus J_{i+v})))) \tag{14}$$

where $W_I = W_f \oplus W_1, W_J = W_b \oplus W_1, W_y = W_f \oplus W_1 \oplus W_b$ and $\oplus$ is the concatenation operator. σ_1 and σ_2 denote sigmoid and softmax activation function, respectively. I_0 and J_0 denote the initial states with constant zero entries. The output of the last softmax layer indicates how likely is it that the nucleotide in the center of the sliding window is coding.

Note that the step interval v must be a multiple of three here. This is because that for a mRNA sequence $\boldsymbol{s}$, we assign 1 to the coding labels y_i if the position i in $\boldsymbol{s}$ belongs to the first nucleotide of a codon in practice, and hence the coding label sequence $\boldsymbol{y}$ is actually not always 1 in open reading frame, but shows a periodicity of three nucleotides (e.g., [1,0,0,1,0,0,...]). In our opinion, assigning labels about first nucleotide of a codon for each sample is biologically significant, which teach more information about codon to neural network. That is, if a nucleotide is the first nucleotide of a codon, it will imply that the next two nucleotides also belongs to coding region. However, given that a nucleotide belongs to coding region, we do not really know which codon it belongs to. The setting of v is significantly relevant to the prediction performance. If it is set small, $\boldsymbol{x}_i$ is almost the same as $\boldsymbol{x}_{i-v}$ and $\boldsymbol{x}_{i+v}$, in which case I_{i-v} and J_{i+v} provide almost no additional information for the prediction $\hat{y}_i$. Otherwise, if it is set large, the condition that label dependency might not be satisfied, and hence we choose the optimal setting by experiment.

3.5. Convolutional Neural Network for Translation Initiation Site prediction

At the first stage of NeuroTIS, the coding score $\hat{\boldsymbol{y}}$ is calculated, and at this stage, we employ a CNN (Combining-CNN) to estimate conditional probability distribution $p(z_\kappa|\hat{\boldsymbol{y}}_{\kappa,r_1}, \boldsymbol{s}_{\kappa,r_1}, \hat{\boldsymbol{y}}_{\kappa',r_1})$. CNN Lecun et al. (2015) is a specilized feedforward neural network which has been successfully applied to many applications such as image, video, speech and language processing Du, Xiong, Wu, Zhang, Zhang and Tao (2017); Peng, Zhu, Feng, Shen, Zhang and Zhou (2019); Liu, Lu, He, Zhang and Chen (2017); Abdel-Hamid, Mohamed, Jiang, Deng, Penn and Yu (2014); Kim (2014). It is characterised by the presence of convolutional layers which use a stack of convolutional kernels to detect local patterns that may occur in different positions. Typically, a CNN consists of an input layer, multiple pairs of convolutional-pooling layers, one or more fully connected layers and the last softmax layer. In contrast, Combining-CNN receives two inputs, which separates two kinds of features by feed them into additional univariate networks summed at the last softmax layer. The basic network architecture of Combining-CNN is shown in Table 1.

As shown in Table 1, it is observed that apart from TIS sequence $\boldsymbol{s}_{\kappa,r_1}$, Combining-CNN receives extra 618-dimensional features at the first fully connected layer, including 602-dimensional coding scores ($\hat{\boldsymbol{y}}_{\kappa,r_1}, \hat{\boldsymbol{y}}_{\kappa',r_1}$) and

Table 1
Different network architectures in experiments.

ID	Network architecture	Coding-ANN	Combining-CNN	
0	input layer	64	618	301 x 4
1	conv layer	-	-	100 x (6,4)
2	maxpool layer	-	-	(2,1)
3	dropout layer	-	-	0.3
4	conv layer	-	-	100 x (7,1)
5	maxpool layer	-	-	(2,1)
6	dropout layer	-	-	0.3
7	dense layer	5	20	
8	dropout layer	0.2	0.3	
9	softmax layer	2	2	

Note: - means empty

Table 2
The 16 global features utilized in NeuroTIS.

ID	Global features
1	the length of the upstream sequence to an ATG;
2	the length of the downstream sequence to an ATG;
3	the log ratio of (2) to (1);
4	the number of upstream ATGs to an ATG;
5	the number of downstream ATGs to an ATG;
6	the log ratio of (5) to (4);
7	the number of in-frame upstream ATGs to an ATG;
8	the number of in-frame downstream ATGs to an ATG;
9	the log ratio of (8) to (7);
10	the number of upstream stop codons from an ATG;
11	the number of downstream stop codons from an ATG;
12	the log ratio of (11) to (10);
13	the number of in-frame upstream stop codons from an ATG;
14	the number of in-frame downstream stop codons from an ATG;
15	the log ratio of (14) to (13);
16	the length of the open reading frame starting at an ATG.

16-dimensional global features utilized in Li et al. (2005). Table 2 shows all the 16 global features. Just as TISRover can easily capture local features such as Kozak consensus motifs and codon interactions by utilizing a CNN, Combining-CNN preserves the characteristic of TISRover by directly feeding TIS sequence into convolutional layers. The difference lies in that it enhances the power of exploiting coding features by explicitly feeding coding scores around start and stop codon into the first fully connected layer. Moreover, TISRover only receives a fixed length of TIS sequence, which cannot exploit the whole sequence to learn global features, whereas Combining-CNN feed explicit global features into the fully connected layer of CNN. With the fusion of multiple relevant features, Combining-CNN jointly infer TIS in an effective way.

It is worth noting that the network architecture of Combining-CNN is not new, it has emerged in recent works. Ghafoorian, Karssemeijer, Heskes, van Uden, Sanchez, Litjens, de Leeuw, van Ginneken, Marchiori and Platel (2017) proposes several deep CNN architectures that take explicit location features into CNN for segmentation of white matter hyperintensities in brain MR images, they observe that the CNNs that incorporate location information substantially outperform a conventional segmentation method that do not integrate location information. Zhao, Yang, Luo, Lin and Wang (2016) presents a syntax CNN based drug-drug interaction extraction method. A better performance is obtained when the position and part of speech feature are introduced to extend the embedding of each word. The above applications demonstrate the promise of incorporating domain knowledge into CNN.

Table 3
The average auROC scores of Coding-BRNN1 and Coding-BRNN2 on Human and Mouse datasets when choosing different parameter υ.

The setting of υ	3	30	57	84	111	138
Human	.9721	.9871	.9931	.9951	.9941	.9935
Mouse	.9725	.9872	.9932	.9953	.9953	.9942

Table 4
Performance comparison of Coding-ANN, Coding-BRNN1 and Coding-BRNN2 on Human and Mouse datasets.

Methods	Human		Mouse	
	auROC	auPRC	auROC	auPRC
Coding-ANN	.9717	.9694	.9723	.9713
Coding-BRNN1	.9950	.9941	.9952	.9948
Coding-BRNN2	.9951	.9945	.9954	.9950

4. Experiments

In this section, we conduct four experiments on two benchmark gene datasets. The first is to choose optimal step interval υ for Coding-BRNN. The second is to evaluate the performance of Coding-BRNN for coding region prediction. In the third experiment, we make a comparison of NeuroTIS to existing state-of-the-art methods such as DIANA-TIS Hatzigeorgiou (2002), GMM Li et al. (2005), iTIS-PseTNC Chen et al. (2014), TITER Zhang et al. (2017) and TISRover Zuallaert et al. (2018b). The last experiment is to evaluate the time cost of NeuroTIS.

4.1. Datasets

The datasets used in this paper are Human and Mouse datasets downloaded from Refseq (`ftp://ftp.ncbi.nih.gov/refseq/`). A total number of 24842 and 19900 transcripts are obtained after clean up procedure, respectively. All these transcripts have canonical TISs. We adopt the hold-out strategy, randomly selecting 20000 and 15000 transcripts as training set to test the remaining 4282 and 4900 ones, for Human and Mouse datasets, respectively. TISs are selected according to a fixed ratio of positive to negative samples (1:1 for balanced case and 1:5 for imbalanced case).

4.2. Performance measurements

In order to evaluate the performance of NeuroTIS, the analysis in this paper employs two evaluation criteria in terms of area under the Receiver Operating Characteristic Curve (auROC) and Precision Recall Curve (auPRC). ROC curves are commonly used to measure performance for binary classification problems, whereas PRC is a better measure when dealing with an unbalanced dataset. The two criteria are both based on the notions of TP, FP, TN, and FN, which correspond to number of true positives, false positives, true negatives, and false negatives. In a ROC, one typically plots the true positive rate (TPR=TP/(TP+FN)) as a function of the false negative rate (FNR=FN/(FN+TN)), and in a PRC, one plots the precision (TP/(TP+FP)) as a function of the recall (TPR). The auROC and auPRC can be calculated by using the trapezoidal areas created between each ROC and PRC points, respectively. Further details can be found in Mitchell, Carbonell and Michalski (1997); Davis and Goadrich (2006). Moreover, we also perform the Delong's test Delong, Delong and Clarkepearson (1988) which is a non-parametric approach to the significance analysis of areas under two ROC curves.

4.3. The choice of step interval υ

In order to choose an optimal step interval υ for Coding-BRNN, we test on Human and Mouse datasets, and calculate the average auROC scores of Coding-BRNN1 and Coding-BRNN2 by gradually increasing the step interval υ. The network architectures are shown in Fig. 4, where weight matrices W_1, W_2, W_f, W_b are taken as $64\times5, 5\times2, 5\times5, 5\times5$ and $64\times5, 5\times2, 2\times5, 2\times5$ for Coding-BRNN1 and Coding-BRNN2, respectively. The r_2 here is set 42. As shown in Table 3, prediction performance of Coding-BRNN improves along with the increase of υ, and decreases when υ is larger than 84. The underlying reason could be that $I_{i-\upsilon}$ and $J_{i+\upsilon}$ provide more and more information for the

Table 5
Performance comparison of NeuroTIS with the other state-of-the-art methods on Human dataset.

Human	Balanced		Imbalanced	
	auROC	auPRC	auROC	auPRC
DIANA-TIS	.9409	.9615	.9413	.8977
GMM	.9517	.9398	.9524	.7926
iTIS-PseTNC	.8684	.8091	.7923	.5492
TITER	.9133	.9203	.9143	.7655
TISRover	.9811	.9802	.9887	.9553
NeuroTIS1	.9937	.9937	.9943	.9792
NeuroTIS2	**.9987**	**.9987**	**.9985**	**.9928**

Table 6
Performance comparison of NeuroTIS with the other state-of-the-art methods on Mouse dataset.

Mouse	Balanced		Imbalanced	
	auROC	auPRC	auROC	auPRC
DIANA-TIS	.9460	.9620	.9475	.8989
GMM	.9505	.9417	.9501	.7818
iTIS-PseTNC	.8205	.7683	.8017	.5678
TITER	.9081	.9168	.9079	.7547
TISRover	.9781	.9759	.9891	.9551
NeuroTIS1	.9948	.9940	.9962	.9867
NeuroTIS2	**.9984**	**.9983**	**.9985**	**.9920**

Table 7
Brief description of time cost on balanced/imbalanced Human and Mouse datasets with regard to NeuroTIS.

Dataset	Coding Number	TISs Number	Time cost (min)	
			Coding-BRNN	Combining-CNN
Human	2,826,820	49,684/149,052	45	16/55
Mouse	2,329,978	39,800/119,400	40	13/42

prediction $\hat{y}_i$ along with the increase of v, and the information reaches the highest level when v is 84 ($\boldsymbol{x}_i$, $\boldsymbol{x}_{i-v}$ and $\boldsymbol{x}_{i+v}$ are completely different).

4.4. Performance comparison of Coding-ANN, Coding-BRNN1 and Coding-BRNN2

We make a comparison of Coding-ANN Hatzigeorgiou (2002); Tzanis et al. (2012), Coding-BRNN1 and Coding-BRNN2. The network architecture of Coding-ANN is shown in Table 2. To avoid the imbalance data problem, negative examples are chosen such that their number equals that of the positive examples. In practice, we only randomly select a small ratio of the whole samples and we find that all the three methods work well, which proves that coding usage statistic is a very robust feature to identify coding region in mRNA sequences. Moreover, they also converge quickly due to small-scale network architecture. The experimental results are shown in Table 4 and Fig. 6, we can see that both of two BRNN-based methods achieve comparable performance for coding region prediction on human and mouse datasets, their achieved auROC and auPRC scores only show a very slight difference, and the p-value of Delong's test between them is larger than 0.06 (-Log10(p-value)<1.2218). Furthermore, the two methods both show much better prediction performance than Coding-ANN, the p-value of each one against Coding-ANN is smaller than 1e-4 (-Log10(p-value)>4), which proves the significance of exploiting statistical dependency in mRNA sequences for coding region prediction.

4.5. Performance comparison with existing state-of-the-art methods

We compare the performance of NeuroTIS with that of existing state-of-the-art methods such as DIANA-TIS Hatzigeorgiou (2002), GMM Li et al. (2005), iTIS-PseTNC Chen et al. (2014), TITER Zhang et al. (2017) and TISRover

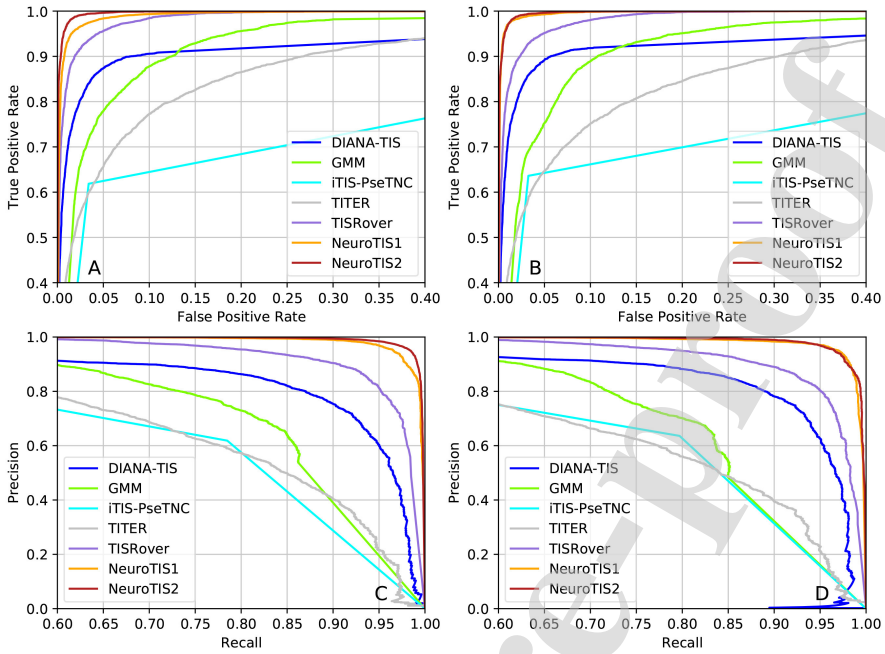


Figure 5: Comparison results of our model, TISRover, TITER, iTIS-PseTNC, GMM and DIANA-TIS. (A) the ROC curve on Human dataset; (B) the ROC curve on Mouse dataset; (C) the precision-recall curve on Human dataset; (D) the precision-recall curve on Mouse dataset.

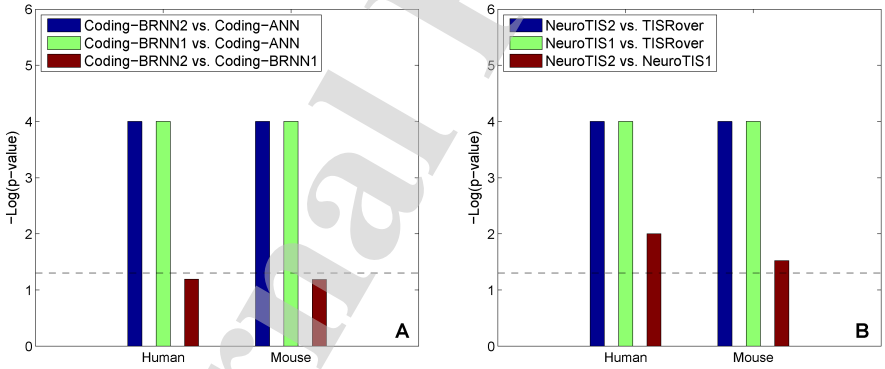


Figure 6: Delong's test between distinct methods on Human and Mouse datasets. (A) Significance with -Log10(p-value) between Coding-BRNN2, Coding-BRNN1 and Coding-ANN; (B) Significance with -Log10(p-value) between NeuroTIS2, NeuroTIS1 and TISRover. Dashed lines represent the -log10(p-value = 0.05) threshold.

Zuallaert et al. (2018b). All the methods are trained and evaluated with the same dataset for fair comparison. From Table 5-6, it is observed that NeuroTIS performs the best among the existing methods and achieves the highest auROC, auPRC scores on human and mouse datasets, both of the auROC and auPRC scores exceed .998 in balanced case, and .998, .992 in imbalanced case, respectively. In particular, as the imbalancedness degree increases from 1:1 to 1:5, the auPRC score of the existing methods drop sharply whereas NeuroTIS decreases slowly (from .998 to .992), which indicates that NeuroTIS can effectively reduce false positives. This is owing to its inclusion of evidence contributed from multiple relevant biological features. We also plot the ROC and precision-recall curves on imbalanced human and mouse datasets. As it can be seen in Fig. 5, given a fixed false positive rate of 2.5%, NeuroTIS respectively achieves a sensitivity of 99.08% and 99.09% on human and mouse dataset, a 7.42%, 7.65% improvement

over the second best method, TISRover. Delong's test between NeuroTIS and TISRover yields p-value smaller than 1e-4 (-Log10(p-value)>4) both on human and mouse datasets as shown in Fig. 6, which proves that the performance improvement of NeuroTIS over TISRover in auROC is statistically significant. As for PRC shown in Fig. 5, given a fixed recall of 97.5%, NeuroTIS respectively achieve a precision of 93.15% and 92.65% on human and mouse dataset, a 24.03% and 30.12% improvement over TISRover. Moreover, in order to evaluate the effectiveness of coding features and global features, we conduct an ablation study to append 602 (corresponding to NeuroTIS1) and 618 (corresponding to NeuroTIS2) features to the first fully-connected layer of Combining-CNN, respectively. Note that NeuroTIS1 share almost the same network architectures with TISRover, except extra coding features. As it can be seen from Table 5-6 and Fig. 6, NeuroTIS1 has a significant prediction performance improvement over TISRover on human and mouse datasets, the average auROC and auPRC scores improved by 1.05%, 1.58% in balanced case, and 0.64%, 2.78% in imbalanced case, and the p-value of Delong's test between NeuroTIS1 and TISRover are both smaller than 1e-4 (-Log10(p-value)>4) on human and mouse datasets. Meanwhile, NeuroTIS2 yields a performance improvement over NeuroTIS1 from Table 5 and Fig. 6, especially in human dataset, the auPRC score improved by 1.36% in imbalanced case, and the Delong's test between NeuroTIS2 and NeuroTIS1 yields p-value smaller than 0.01 (-Log10(p-value)>2) and 0.03 (-Log10(p-value)>1.5229) on human and mouse dataset, respectively. All the results demonstrate that NeuroTIS is a high-accuracy TIS prediction method.

4.6. Time cost of NeuroTIS

We further briefly analyze the computational cost of NeuroTIS. Table 7 gives the time cost of NeuroTIS on Human and Mouse datasets. All the experiments are conducted on an Intel Core i5-7400 CPU 3.00GHz PC with 16GB RAM. Coding-BRNN and Combining-CNN are implemented in Tensorflow. As shown in Table 7, we can see that larger dataset requires more additional training time. Moreover, the time cost of TISRover is basically equivalent to Combining-CNN and not mentioned here, NeuroTIS has one more stage compared with TISRover, and thus it requires more time to train and test Coding-BRNN.

5. Discussions

The relevant features for TISs prediction are consensus motifs, coding features and global features, among which it is known that the most prominent features is the coding features. How to extract these three kinds of features fully from uncharacterized genomic sequences is the core problem that a successful TISs predictor needs to address. Consensus motifs are easy to capture from genomic sequences by CNN because it seems like local patterns, whereas coding features are more difficult to learn for its higher-order distant interactions among nucleotides. In this article, we propose a hybrid framework called NeuroTIS for TISs prediction, which *explicitly* model label dependencies and combines the three kinds of features in a CNN. Evaluated on two benchmark datasets, NeuroTIS yields remarkable prediction performance when compared with five state-of-the-art methods.

In our work, there are three potential factors which may explain the outstanding performance in predicting TISs. First, dependency network builds a natural and explicit representation of label dependency and considering the label dependencies among coding region, between coding region and TIS, makes the prediction more sensible. Second, BRNN effectively and automatically learn the label dependency among coding region, and significantly improve the performance of coding region prediction. Third, coding features and global features, which contain explicit biological information, e.g., codon usage, the scanning model Kozak (1989), enhances the prediction performance of neural networks.

In spite of remarkable prediction performance, NeuroTIS still has some limitations. First, it requires full-length mRNA sequences to generate the global features. Fortunately, more and more full-length mRNA sequences are available now. Second, it only consider canonical downstream TISs in mRNAs, but there exists multiple alternative TISs (including both AUG and non-AUG codons) in many mRNAs Zhang et al. (2017). Third, it assumes a directed dependency between TIS and coding region for efficient inference. However, it might benefit from a cyclic dependency. Future studies on these points are warranted.

6. Conclusion

In conclusion, we present NeuroTIS, a hybrid dependency network and deep learning framework to infer translation initiation sites in mRNA sequences, which *explicitly* model label dependencies among coding region, between

coding region and translation initiation site via a dependency network. Moreover, a Bidirectional Recurrent Neural Network and a Convolutional Neural Network are employed for feature learning and approximate inference, which makes it efficient and effective for translation initiation sites prediction. The experiments results show that NeuroTIS significantly outperform existing state-of-the-art methods.

Acknowledgments

This work is supported by the Natural Science Foundation of China under Grants 11674352 and 91853123.

References

Abdel-Hamid, O., Mohamed, A.R., Jiang, H., Deng, L., Penn, G., Yu, D., 2014. Convolutional neural networks for speech recognition. IEEE/ACM Transactions on Audio Speech & Language Processing 22, 1533–1545.

Alipanahi, B., Delong, A., Weirauch, M.T., Frey, B.J., 2015. Predicting the sequence specificities of dna-and rna-binding proteins by deep learning. Nature biotechnology 33, 831–838.

Barbosa, C., Peixeiro, I., Romão, L., 2013. Gene expression regulation by upstream open reading frames and human disease. PLoS genetics 9.

Bernal, A., Crammer, K., Hatzigeorgiou, A., Pereira, F., 2007. Global discriminative learning for higher-accuracy computational gene prediction. PLoS computational biology 3, e54.

Boersma, S., Khuperkar, D., Verhagen, B.M., Sonneveld, S., Grimm, J.B., Lavis, L.D., Tanenbaum, M.E., 2019. Multi-color single-molecule imaging uncovers extensive heterogeneity in mrna decoding. Cell 178, 458–472.

Brunak, S., ., Engelbrecht, J., ., Knudsen, S., ., 1991. Prediction of human mrna donor and acceptor sites from the dna sequence. Journal of Molecular Biology 220, 49–65.

Chen, W., Feng, P.M., Deng, E.Z., Lin, H., Chou, K.C., 2014. itis-psetnc: A sequence-based predictor for identifying translation initiation site in human genes using pseudo trinucleotide composition. Analytical Biochemistry 462, 76–83.

Davis, J., Goadrich, M., 2006. The relationship between precision-recall and roc curves, in: Proceedings of the 23rd international conference on Machine learning, pp. 233–240.

Delong, E.R., Delong, D.M., Clarkepearson, D.L., 1988. Comparing the areas under two or more correlated receiver operating characteristic curves: A nonparametric approach. Biometrics 44, 837–845.

Dinarelli, M., Tellier, I., 2016. Improving recurrent neural networks for sequence labelling .

Du, B., Xiong, W., Wu, J., Zhang, L., Zhang, L., Tao, D., 2017. Stacked convolutional denoising auto-encoders for feature representation. IEEE Transactions on Cybernetics 47, 1017–1027.

Du, X., Yao, Y., Diao, Y., Zhu, H., Zhang, Y., Li, S., 2018. Deepss: Exploring splice site motif through convolutional neural network directly from dna sequence. IEEE Access 6, 32958–32978.

Elman, J.L., 1990. Finding structure in time. Cognitive Science 14, 179–211.

Errattahi, R., El Hannani, A., Salmam, F.Z., Ouahmane, H., 2019. Incorporating label dependency for asr error detection via rnn. Procedia computer science 148, 266–272.

Fickett, J.W., Tung, C.S., 1992. Assessment of protein coding measures. Nucleic Acids Research 20, 6441–50.

Geman, S., 1984. Stochastic relaxation, gibbs distributions, and the bayesian restoration of images. IEEE Transactions on Pattern Analysis & Machine Intelligence 6.

Ghafoorian, M., Karssemeijer, N., Heskes, T., van Uden, I.W., Sanchez, C.I., Litjens, G., de Leeuw, F.E., van Ginneken, B., Marchiori, E., Platel, B., 2017. Location sensitive deep convolutional neural networks for segmentation of white matter hyperintensities. Scientific Reports 7, 1–12.

Goodfellow, I., Bengio, Y., Courville, A., 2016. Deep learning. MIT press.

Guo, Y., Gu, S., 2011. Multi-label classification using conditional dependency networks, in: IJCAI 2011, Proceedings of the 22nd International Joint Conference on Artificial Intelligence, Barcelona, Catalonia, Spain, July 16-22, 2011.

Hatzigeorgiou, A., Mache, N., Reczko, M., 1996. Functional site prediction on the dna sequence by artificial neural networks, in: Proceedings IEEE International Joint Symposia on Intelligence and Systems, IEEE. pp. 12–17.

Hatzigeorgiou, A.G., 2002. Translation initiation start prediction in human cdnas with high accuracy. Bioinformatics 18, 343–350.

Heckerman, D., Chickering, D.M., Meek, C., Rounthwaite, R., Kadie, C., 2013. Dependency networks for collaborative filtering and data visualization. Journal of Machine Learning Research 1, 49–75.

Hinnebusch, A.G., Ivanov, I.P., Sonenberg, N., 2016. Translational control by 5'-untranslated regions of eukaryotic mrnas. Science 352, 1413–1416.

Jordan, M.I., 1986. Serial order: a parallel distributed processing approach. technical report, june 1985-march 1986 .

Khuperkar, D., Hoek, T.A., Sonneveld, S., Verhagen, B.M., Boersma, S., Tanenbaum, M.E., 2020. Quantification of mrna translation in live cells using single-molecule imaging. Nature Protocols , 1–28.

Kim, Y., 2014. Convolutional neural networks for sentence classification. arXiv preprint arXiv:1408.5882 .

Koller, D., Friedman, N., 2009. Probabilistic Graphical Models: Principles and Techniques - Adaptive Computation and Machine Learning.

Kozak, M., 1983. Translation of insulin-related polypeptides from messenger rnas with tandemly reiterated copies of the ribosome binding site. Cell 34, 971–978.

Kozak, M., 1989. The scanning model for translation: an update. The Journal of cell biology 108, 229–241.

Lecun, Y., Bengio, Y., Hinton, G., 2015. Deep learning. Nature 521, 436.

Li, G., Leong, T.Y., Zhang, L., 2005. Translation initiation sites prediction with mixture gaussian models. IEEE Transactions on Knowledge & Data Engineering 17, 1152–1160.

Li, H., Jiang, T., 2004. A class of edit kernels for svms to predict translation initiation sites in eukaryotic mrnas., in: Eighth International Conference on Resaerch in Computational Molecular Biology.
Li, J., Liu, H., Wong, L., Yap, R.H., 2004. Techniques for recognition of translation initiation sites, in: The Practical Bioinformatician. World Scientific, pp. 71–89.
Liu, Q., Lu, X., He, Z., Zhang, C., Chen, W.S., 2017. Deep convolutional neural networks for thermal infrared object tracking. Knowledge-Based Systems 134, 189–198.
Mesnil, G., He, X., Deng, L., Bengio, Y., 2013. Investigation of recurrent-neural-network architectures and learning methods for spoken language understanding., in: Interspeech, pp. 3771–3775.
Min, X., Zeng, W., Chen, S., Chen, N., Chen, T., Jiang, R., 2017. Predicting enhancers with deep convolutional neural networks. BMC bioinformatics 18, 478.
Mitchell, T.M., Carbonell, J.G., Michalski, R.S., 1997. Machine Learning.
Nishikawa, T., Ota, T., Isogai, T., . Prediction of fullness of cdna fragment sequences by combining statistical information and similarity with protein sequences .
Pearl, J., 1986. Fusion, propagation, and structuring in belief networks. Artificial intelligence 29, 241–288.
Pedersen, A.G., Nielsen, H., 1997. Neural network prediction of translation initiation sites in eukaryotes: perspectives for est and genome analysis., in: International Conference on Intelligent Systems for Molecular Biology.
Pelletier, J., Sonenberg, N., 1988. Internal initiation of translation of eukaryotic mrna directed by a sequence derived from poliovirus rna. Nature 334, 320–325.
Peng, X., Zhu, H., Feng, J., Shen, C., Zhang, H., Zhou, J.T., 2019. Deep clustering with sample-assignment invariance prior. IEEE Transactions on Neural Networks and Learning Systems .
Pham, D.T., Karaboga, D., 1999. Training elman and jordan networks for system identification using genetic algorithms. Artificial Intelligence in Engineering 13, 107–117.
Rajapakse, J.C., Ho, L.S., 2005. Markov encoding for detecting signals in genomic sequences. IEEE/ACM Transactions on Computational Biology and Bioinformatics 2, 131–142.
Read, J., Martino, L., Hollmen, J., 2017. Multi-label methods for prediction with sequential data. Pattern Recognition 63, 45–55.
Saeys, Y., 2004. Feature selection for classification of nucleic acid sequences. Ph.D. thesis. Ghent University.
Salamov, A.A., 1998. Assessing protein coding region integrity in cdna sequencing projects. Bioinformatics 14, 384.
Schum, D.A., 1994. The Evidential Foundations of Probabilistic Reasoning by David A. Schum.
Schuster, M., Paliwal, K.K., 1997. Bidirectional recurrent neural networks. IEEE Transactions on Signal Processing 45, 2673–2681.
Solovyev, V., Umarov, R., 2016. Prediction of prokaryotic and eukaryotic promoters using convolutional deep learning neural networks. arXiv preprint arXiv:1610.00121 .
Sonenberg, N., Hinnebusch, A.G., 2009. Regulation of translation initiation in eukaryotes: Mechanisms and biological targets. Cell 136, 0–745.
Tzanis, G., Berberidis, C., Vlahavas, I., 2007. Mantis: a data mining methodology for effective translation initiation site prediction, in: International Conference of the IEEE Engineering in Medicine & Biology Society.
Tzanis, G., Berberidis, C., Vlahavas, I., 2012. Stacktis: A stacked generalization approach for effective prediction of translation initiation sites. Computers in Biology & Medicine 42, 61–69.
Xi, J., Li, A., Wang, M., 2017. A novel network regularized matrix decomposition method to detect mutated cancer genes in tumour samples with inter-patient heterogeneity. Scientific Reports 7, 2855.
Xi, J., Li, A., Wang, M., 2018. A novel unsupervised learning model for detecting driver genes from pan-cancer data through matrix tri-factorization framework with pairwise similarities constraints. Neurocomputing 296, 64–73.
Yi, Z., Liu, X., Macleod, J.N., Liu, J., 2017. Deepsplice: Deep classification of novel splice junctions revealed by rna-seq, in: IEEE International Conference on Bioinformatics & Biomedicine.
Yu, H., Meng, W., Mao, Y., Zhang, Y., Sun, Q., Tao, S., 2019. Deciphering the rules of mrna structure differentiation in saccharomyces cerevisiae in vivo and in vitro with deep neural networks. RNA biology 16, 1044–1054.
Yuan, X., Bai, J., Zhang, J., Yang, L., Duan, J., Li, Y., Gao, M., 2018. Condel: Detecting copy number variation and genotyping deletion zygosity from single tumor samples using sequence data. IEEE/ACM transactions on computational biology and bioinformatics .
Yuan, X., Zhang, J., Yang, L., Bai, J., Fan, P., 2017. Detection of significant copy number variations from multiple samples in next-generation sequencing data. IEEE transactions on nanobioscience 17, 12–20.
Zhang, S., Hu, H., Jiang, T., Zhang, L., Zeng, J., 2017. Titer: predicting translation initiation sites by deep learning. Bioinformatics 33, i234–i242.
Zhao, Z., Yang, Z., Luo, L., Lin, H., Wang, J., 2016. Drug drug interaction extraction from biomedical literature using syntax convolutional neural network. Bioinformatics 32, 3444–3453.
Zien, A., Rätsch, G., Mika, S., Schölkopf, B., Lengauer, T., Müller, K.R., 2000. Engineering support vector machine kernels that recognize translation initiation sites. Bioinformatics 16, 799.
Zuallaert, J., Godin, F., Kim, M., Soete, A., Saeys, Y., De Neve, W., 2018a. Splicerover: interpretable convolutional neural networks for improved splice site prediction. Bioinformatics 34, 4180–4188.
Zuallaert, J., Kim, M., Soete, A., Saeys, Y., Neve, W.D., 2018b. Tisrover: Convnets learn biologically relevant features for effective translation initiation site prediction. International Journal of Data Mining and Bioinformatics 20, 267–284.

Chao Wei received the B.S. degree in information and computing science from Huanggang Normal University, Huanggang, Hubei, China, in 2012 and the M.E. degree in computer technology from Wuhan University, Wuhan, Hubei, China, in 2014. Currently, he is working towards Ph.D degree in the School of Computer Science and Technology, Xidian University, Xi'an, China. His research interests including machine learning and bioinformatics.

Junying Zhang received the Ph.D. degree in Signal and Information Processing from Xidian University, Xi'an, China, in 1998. From 2001 to 2002,

she was a visiting scholar at the Department of Electrical Engineering and Computer Science, The Catholic University of America, Washington, DC, USA, and in 2007, she was a visiting professor at the Department of Electrical Engineering and Computer Science, Virginia Polytechnic Institute and State University, USA. She is currently a Professor in the School of Computer Science and Technology, Xidian University, Xi'an, China. Her research interests focus on Intelligent Information Processing, including Machine Learning and its application to Bioinformatics.

Xiguo Yuan received the B.S. and M.S. degree in computer applications from Wuhan University of Science & Technology, Wuhan, Hubei, China, in 2005 and 2008, respectively. He received the Ph.D. degree in computer applications from Xidian University, Xi'an, China, in 2011. He is currently an associate professor in the School of Computer Science and Technology, Xidian University.

Zongzhen He received the B.S. degrees in computer science and technology from Luoyang Institute of Science and Technology, Henan, China, in 2011 and the M.S. degree in computer applications from Zhengzhou University, Henan, China, in 2014. Currently, she is working towards Ph.D degree in the School of Computer Science and Technology, Xidian University, Xi'an, China. Her research interests including machine learning and bioinformatics.

Guojun Liu received his B.S. degree in Computer Science and Technology from Xi'an Technological University in 2005, and his M.S. degree in computer software and theory from Taiyuan University of Science and Technology in 2011. Currently, he is working towards Ph.D degree in the School of Computer Science and Technology, Xidian University, Xi'an, China. His research interests including machine learning and bioinformatics.

Jinhui Wu received the B.S. degree in Computer Science and Technology from Shanxi Institute of Technology, YangQuan, Shanxi, China, in 2019. Currently, she is working towards M.E. degree in the School of Computer Science and Technology, Xidian University, Xi'an, China. Her research interests including machine learning and bioinformatics.

Chao Wei: Conceptualization, Methodology, Data curation, Software, Writing-Original draft preparation. **Junying Zhang**: Supervision, Writing-Reviewing and Editing. **Xiguo Yuan:** Writing- Reviewing and Editing. **Zongzhen He**: Writing-Reviewing and Editing. **Guojun Liu**: Writing-Reviewing and Editing. **Jinhui Wu**: Writing-Reviewing and Editing.

Declaration of interests

☒ The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

☐The authors declare the following financial interests/personal relationships which may be considered as potential competing interests: