PIBB 生物化学与生物物理进展
Progress in Biochemistry and Biophysics
2022,49(7):1334~1347
www.pibb.ac.cn

基于卷积神经网络的大肠杆菌启动子预测*

彭宝成[1] 张晓炜[2]** 刘 暘[2] 樊国梁[1]**
([1] 内蒙古大学物理科学与技术学院，呼和浩特 010021；[2] 内蒙古医科大学第一附属医院风湿免疫科，呼和浩特 010050）

摘要 目的 基于位点特异性打分矩阵（position-specific scoring matrices，PSSM）的预测模型已经取得了良好的效果，基于PSSM的各种优化方法也在不断发展，但准确率相对较低，为了进一步提高预测准确率，本文基于卷积神经网络（convolutional neural networks，CNN）算法做了进一步研究。**方法** 采用PSSM将启动子序列处理成数值矩阵，通过CNN算法进行分类。大肠杆菌K-12（*Escherichia coli* K-12，*E.coli* K-12，下文简称大肠杆菌）的Sigma38、Sigma54和Sigma70 3种启动子序列被作为正集，编码（Coding）区和非编码（Non-coding）区的序列为负集。**结果** 在预测大肠杆菌启动子的二分类中，准确率达到99%，启动子预测的成功率接近100%；在对Sigma38、Sigma54、Sigma70 3种启动子的三分类中，预测准确率为98%，并且针对每一种序列的预测准确率均可以达到98%以上。最后，本文以Sigma38、Sigma54、Sigma70 3种启动子分别和Coding区或者Non-coding区序列做四分类，预测得到的准确性为0.98，对3种Sigma启动子均衡样本的十交叉检验预测精度均可以达到0.95以上，海明距离为0.016，Kappa系数为0.97。**结论** 相较于支持向量机（support vector machine，SVM）等其他分类算法，CNN分类算法更具优势，并且基于CNN的分类优势，编码方式亦可以得到简化。

关键词 大肠杆菌，位点特异性打分矩阵，卷积神经网络，多分类
中图分类号 Q61 **DOI**：10.16476/j.pibb.2021.0139

启动子是基因序列中一段可以调控基因表达的核苷酸序列（序列中含有起始位点），控制着基因的表达与否，因此启动子在基因的转录和表达中具有重要的地位。

在大肠杆菌基因组中，启动子是一段分布于起始位点上游60 bp及其下游20 bp、包含起始位点的长度为81 bp的DNA碱基序列。按照Sigma因子的类别，大肠杆菌共有7种启动子，分别是Sigma19、Sigma24、Sigma28、Sigma32、Sigma38、Sigma54、Sigma70。由于启动子序列具有保守性，根据保守性片段区域的不同，启动子又可以分为保守性区域在转录起始位点上游−10至−35位点附近的（以Sigma70为代表）和在转录起始位点上游−12至−24位点附近的（以Sigma54为代表）保守性启动子[1-2]。

大肠杆菌的分布极广，并且作为肠杆菌类的成员经常被作为细菌模式生物而被广泛研究，因此人类对于大肠杆菌的研究非常深入（生物实验方面）。大肠杆菌所有系列的基因序列已经在20世纪被完全测量，但是基于生物实验方法寻找启动子的方式十分耗时、昂贵。虽然它可以较为准确地定位启动子序列，但是在面对海量数据时，效率低的弊端开始凸显，因此计算生物学的研究应运而生。以往的研究者提出了各种各样的模型并且取得了良好的预测效果[3-7]。2015年，丁辉等构建的三联体位置关联矩阵的预测方法，预测Sigma54的精度达到82.0%[8-9]；2015年闫妍等[10]利用位点特异性打分矩阵（positive-specific scoring matrices，PSSM）方法预测Sigma启动子，模型对Sigma54的预测准确率为97.4%，对Sigma38的预测准确率为96.0%，

* 国家自然科学基金（62063024），内蒙古自治区高等学校科学研究项目（NJZY20005）和内蒙古大学大学生创新创业训练计划项目（201912240）资助。
** 通讯联系人。
樊国梁 Tel：0471-4992958，E-mail：fanguoliang@imu.edu.cn
张晓炜 Tel：13948518183，E-mail：13948518183@126.com
收稿日期：2021-05-13，接受日期：2021-10-21

对Sigma70的预测准确率为74.0%。此外还有基于柔性参数+二联体位置关联权重矩阵[11]、位置关联打分特征（position-correlation scoring feature，PCSF）+伪核苷酸特征（pseudo k-tuple nucleotide composition，PseKNC）[12]、启动子元件的位置权重信息+*k*-联体核苷酸频率等方法，后来的趋势表明，研究者趋向于组合多种特征来定义启动子序列[13]，以至于分类特征维数急剧增加，因此不得不在分类时对数据进行降维处理。在算法方面，常用的机器学习算法有支持向量机（support vector machine，SVM）、随机森林、K-近邻、隐马尔科夫、人工神经网络、前向传播算法等或其他算法如线性判别分析、二次判别分析等[14-19]。近些年深度学习算法也逐渐被研究者所关注，并且已有研究者将卷积神经网络（convolutional neural networks，CNN）应用于大肠杆菌启动子Sigma70和非启动子的二分类预测中，对由A（1, 0, 0, 0）、T（0, 1, 0, 0）、G（0, 0, 1, 0）和C（0, 0, 0, 1）编码的向量序列进行预测得到了敏感性S_n（sensitivity）=0.90、特异性S_p（specificity）=0.96、准确率*Acc*（accuracy）=0.84的结果，但准确率仍需进一步提高，编码方式仍需优化[20]。

1　数据集的构建和特征算法

1.1　数据的选取

论文中采用的Sigma38、Sigma54、Sigma70数据集可以从RegulonDB 10.8（http://regulondb.ccg.unam.mx/Downloadable）下载，为了便于后续数据处理，保证数据的一致性，对数据文件做如下处理：

a. 剔除Sigma Factor that recognize the promoter标注为多种类型的行。

b. 删除文件信息（以“#”开头的行）和序列信息列（如“acrDp2”等信息），只留下序列所在列。

c. 删除没有序列内容的行。

经过处理后得到了Sigma38序列146条、Sigma54序列96条、Sigma70序列810条，共计1 052条，作为正集。启动子序列的长度均为81 bp（−60 bp~20 bp，设基因转录起始位点为0）。

分别选取了300条编码区序列和300条基因间序列，长度均为81 bp，作为负集。基因间序列处于两条基因之间的区域，选取中央区域可以最大限度的避免选入启动子序列。

1.2　特征描述

大肠杆菌基因序列为字符序列，而CNN算法（目前为止的大部分算法）能处理的只有数值序列，因此需要将字符序列转换为数值序列，这一步也被称之为特征提取（或采样）[21-22]。在转换过程中丢失的信息越少则对序列的描述也就越精确。首先，每一类序列的训练集分别构建了位点特异性打分矩阵。其次，利用该PSSM对该种类的训练集和测试集打分，从而将字母序列转变为数值序列。

1.2.1　构建位点特异性打分矩阵特征

a. 构建频数矩阵

将序列坐标化（从0～80共81个位点），统计每个坐标上A、G、C、T 4种元素出现的频数，如果存在坐标点上面某种核苷酸出现的次数为0，则需要引入伪计数（即将该位点上所有类型核苷酸的频数加一整数，本文中加1）。最终形成的频数矩阵列名为序列坐标（s1~s81），行名为4种核苷酸（A、G、C、T）。

b. 生成频率矩阵（伪计数频率矩阵）*P*

计算每种核苷酸在该位点上出现的频率，即用该位点上每一种元素的频数除以该位点上所有元素的频数和，其中$x_{i,j}$是在*i*位点上第*j*种元素的频数，$P_{i,j}$是在*i*位点上第*j*种元素的频率：

$$P_{ij}=\frac{x_{ij}}{\sum_{j=1}^{4}x_{ij}} \tag{1}$$

c. 生成对数几率比矩阵*oddratio*：

$$oddratio=\frac{P(x|M)}{P(x|R)} \tag{2}$$

其中$P(x|M)$为核苷酸出现的实际概率，$P(x|R)$为核苷酸出现的随机概率（即0.25），因此将频率矩阵*P*中的每个频率值除以0.25即可得到*oddratio*。随后对*oddratio*求以2为底的对数，该矩阵即为位点特异性打分矩阵，矩阵的大小为（4×81）维。

1.2.2　对特定序列打分

从PSSM矩阵中查出特定位点上核苷酸的分值，对给定序列赋值。实现方式是矩阵点乘，最终可以得到1×81或者4×1×81的矩阵（即一维矩阵或者四维矩阵）（图1）。

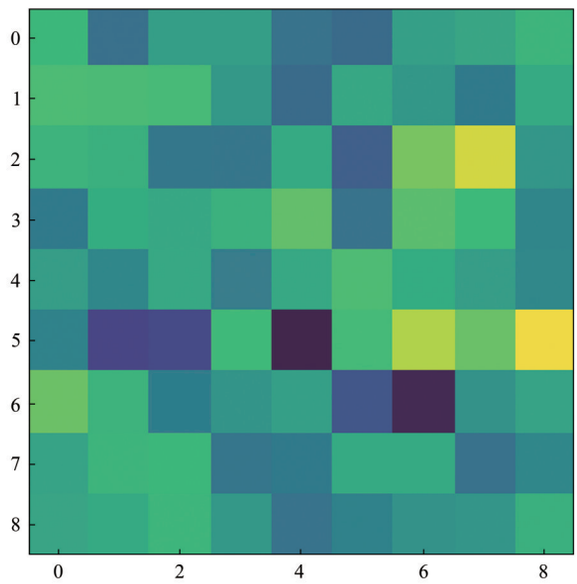

Fig. 1　Sigma70 sequence matrix pixel matrix bitmap after scoring by PSSM

这里的维度指的是通道数：RGB图片有3个通道（R、G、B）即通道数为3，每个维度上的每个位点的数值即为该通道在该位点上的通道颜色浓度，取值范围为0~255，一张RGB图片数值化后就是一个三维数值矩阵（图2）。

类似的，可以把A、G、C、T作为4个通道，将序列数值化的方法就变得容易理解：一条序列在A通道（A通道长度为81）上每个位点的值可以是0（该位点不为A）或者1（该位点为A），对其他3个通道做同样处理，结果如图3c所示。计算证实训练上述方法得到的01数值序列也可以获得好的效果，但是01离散数字序列携带的只有位置分布信息，信息量过小，因此过拟合的现象常常发生。

把A通道中的1替换成PSSM中的打分值。在位点s1上，“g”的打分值为−0.497 5，那么G通道中位点s1上的1就可以被替换为−0.497 5，如果是0则不需要替换（表1）。这种方法显然可以让数值化序列带有更多的信息，对其他通道做同样处理，就构造出了4个维度的数值矩阵（图3d）。

Table 1　Sigma70 position-specific scoring matrix（PSSM）for a certain training set

	s1	s2	……	s80	s81
a	0.273 018	0.273 018	……	0.544 321	0.273 018
g	−0.497 5	−1.125 53	……	−0.337 03	−0.415 04
c	−0.497 5	−0.337 03	……	−0.337 03	−0.061 4
t	0.459 432	0.624 491	……	−0.061 4	0.115 477

图片上每个色块的颜色是由3个通道颜色组合而成的，与此不同的是，启动子序列的每个位点只有1个通道的值（AGCT中的一个）（图3b），因此每个通道中依旧有很多的0存在（图3d），所以可以将4个通道对应位点的数值加和，得到一个没有‘0’的长度为81的向量（图3e），将向量变换成9×9的矩阵绘图以后形成的点阵图片（图1）。依旧是由于序列和图片的不同点，序列的四维数值信息和一维数值信息没有本质上的区别，反映在模型表现上就是预测的准确性没有大的差别。

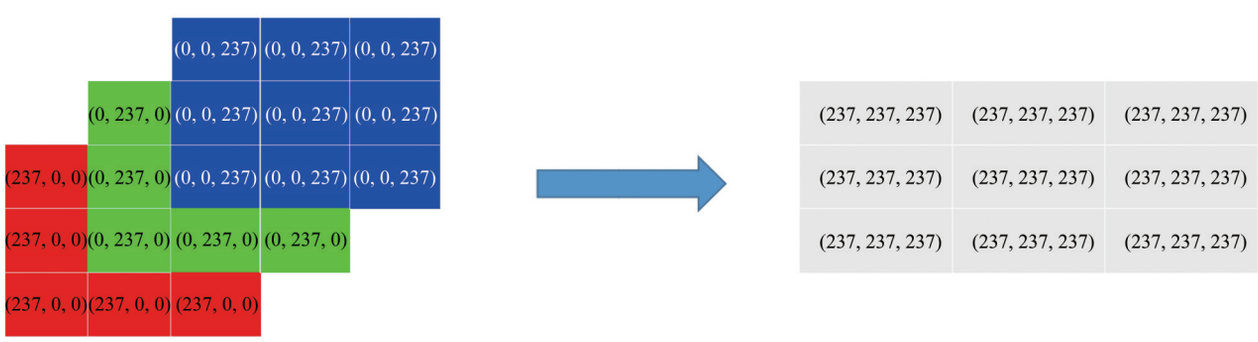

Fig. 2　Picture of RGB three-channel schematic diagram

(a)	G	C	C	T	.....	T	C	G	T
(b)	s1	s2	s3	s4	.....	s78	s79	s80	s81
a	(void)	(void)	(void)	(void)	.....	(void)	(void)	(void)	(void)
g	G	(void)	(void)	(void)	.....	(void)	(void)	G	(void)
c	(void)	C	C	(void)	.....	(void)	C	(void)	(void)
t	(void)	(void)	(void)	T	.....	T	(void)	(void)	T

(c)	To 01 matrix								
	s1	s2	s3	s4	.....	s78	s79	s80	s81
a	0	0	0	0	.....	0	0	0	0
g	1	0	0	0	.....	0	0	1	0
c	0	1	1	0	.....	0	1	0	0
t	0	0	0	1	.....	1	0	0	1

(e)	4D To 1D								
	s1	s2	s3	s4	.....	s78	s79	s80	s81
	−0.497 5	−0.337	−0.058 9	−0.061 4	.....	0.000 00	−0.192 7	−0.337	−0.115 5

(d)	Scoring with PSSM (4D)								
	s1	s2	s3	s4	.....	s78	s79	s80	s81
a	0	0	0	0	.....	0	0	0	0
g	−0.5	0	0	0	.....	0	0	−0.337	0
c	0	−0.337	−0.058 9	0	.....	0	−0.193	0	0
t	0	0	0	−0.06	.....	0.000 00	0	0	−0.115 5

Fig. 3　Overview of conversion method for certain Sigma70 sequence

与图片信息相同的是，启动子的保守性区域可以类比为被人类所理解的图像信息（如数字“1”反映在图像上是几个白灰色像素块组合，图4），无论是保守性序列还是数字“1”都具有不变性，这为研究者们寻找启动子的独有特征奠定了理论基础[23]。

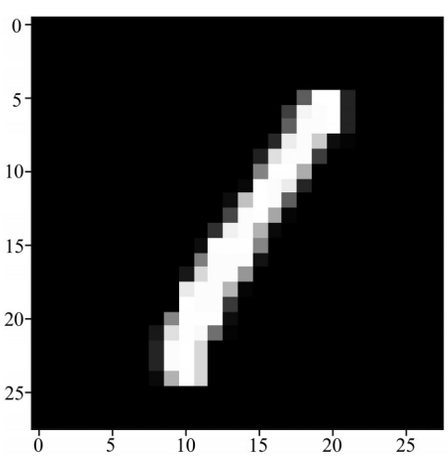

Fig.4　Handwritten numbers “1” from http://deeplearning.net/data/mnist/

1.3　分类算法

CNN是一个兼顾全连接和卷积取样的前馈神经网络算法，它的提出来源于对生物的视觉皮层研究。它在处理多层网格结构数据方面具有巨大优势，因此现在CNN被广泛应用于图像识别、语音识别和自然语言处理等领域。CNN的实现分为两个主要步骤：a. 数据的预处理：将图像等信息数字化和去噪等操作；b. 特征提取和分类：这一部分由CNN网络结构来实现，基础的网络结构包括卷积层、池化层和全连接层3部分，深度CNN是基础网络结构的多层叠加，这样可以实现更多特征的提取从而提高识别精确度。

CNN的主要特点有局部区域连接、权值共享和降采样。a. 局部区域连接（图5左），即前后两层网络的所有神经元并不是都互相连接的，目的是为了模拟视觉神经的选择性聚焦，这有利于减少训练参数；b. 权重共享，即一个卷积核在提取特征时的权重在整张图片上都相同，这意味着一个卷积核只能提取一种特征；c. 降采样，主要有最大值、平均值（图5右）和方均值保留等方式，通过池化层

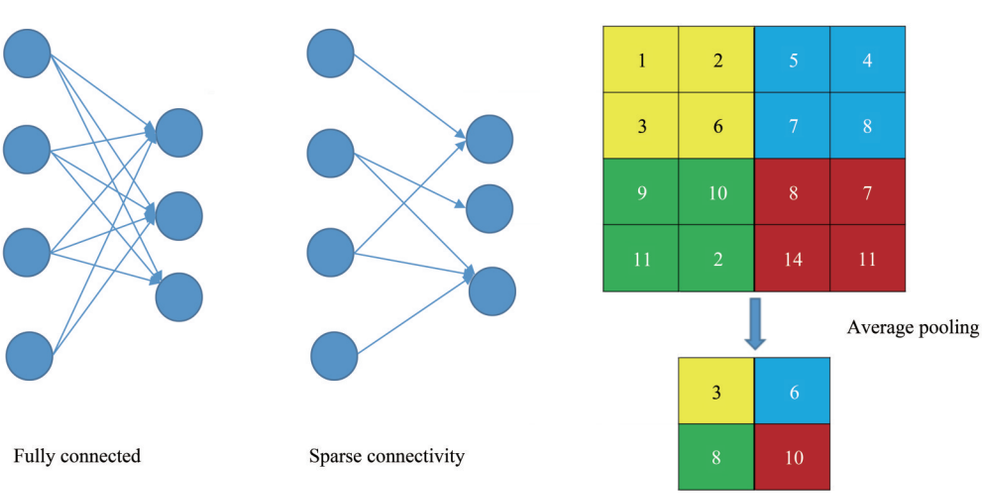

Fig. 5　Fully connected layer and average pooling layer

（pooling layer）实现，目的是为了降低特征分辨率和减少过拟合风险。由于这3个特点，CNN在抽取特征时更能够聚焦图片的重要信息（图4中黑色的背景显然就不是有效信息，因此CNN就不会着重关注）。由于启动子序列的类图片特点，将CNN作为特征提取和分类的算法是比较合适的选择。

2 检验方法

2.1 模型评价

2.1.1 自洽验证

分别从各类序列中抽取96条序列组成均衡样本；构建均衡样本中每种序列的位点特异性打分矩阵（**1.2**节），并按照**1.2**节方式分别使用各自的位点特异性打分矩阵将各自的所有序列（包括均衡样本序列）转变成数值序列。以均衡样本序列作为训练集，采用全部序列测试模型性能。

2.1.2 独立检验

所有序列被分成两部分（数量均等），一份用做训练集，另一部分用作测试集，用训练集的位点特异性打分矩阵将测试集和训练集转变为数值化序列，使用训练集训练模型，测试集测试模型性能。

2.1.3 十交叉检验

该方法与前述独立检验不同的是，构建位点特异性打分矩阵所使用的序列数目不同。十交叉检验随机将数据集分成10份，取其中的9份构建位点特异性打分矩阵并被转换成数值序列后用作训练集，剩余1份被转换成数值序列后用作测试模型性能。如此循环，则共产生了10份测试结果，对10份结果取平均就得到了最终的验证结果。

2.2 验证参数

2.2.1 ROC曲线

ROC曲线即受试者操作特性曲线（receiver operating characteristic curve，常称ROC曲线），描述的是在不同的标准下（主要是阈值），模型的击中率和误报率之间的函数关系。ROC曲线常用在二分类模型当中，但是在多分类模型中也可以采取此参数（将被求ROC参数的样本设为正集，其余种类为负集）。

曲线下面积（area under curve，AUC）一般在0.5~1之间，AUC越大即代表模型的性能越好。ROC曲线的优势在于当正负样本数量对比发生变化时候曲线的形状不会变化。

2.2.2 S_n、S_p、*Acc*

S_n、S_p常用在二分类模型验证当中，S_n表示正确预测正集样本的概率（式3），S_p表示正确预测负集样本的概率（式4）。

$$S_n = \frac{Tp}{Tp + Fn} \tag{3}$$

$$S_p = \frac{Tn}{Tn + Fp} \tag{4}$$

*Acc*无论是在多分类还是二分类都比较常用的参数，它表示样本被正确预测的概率：

$$Acc = \frac{Tn + Tp}{Tn + Fp + Tp + Fn} \tag{5}$$

每个种类序列的准确率*Acc*公式为：

$$Acc = \frac{T}{T + P} \tag{6}$$

在*Acc*的计算公式中，*T*n和*T*p分别表示被预测成功的负集样本数和正集样本数，*F*p和*F*n则表示被预测失败的负集样本数和正集样本数。

在*Acc*的计算公式中，*T*代表该类序列中被预测正确的数目，*P*代表该序列中被预测失败的数目。

3 结果评估

本文采用PSSM矩阵作为识别序列的特征，用打分的方式将字符序列转变成数值序列。基于实际需要，本文做两组四分类：Sigma序列和编码（Coding）序列、Sigma序列和非编码（Non-coding）序列。

3.1 损失函数

模型训练中使用的是交叉熵损失：

$$H(p,q) = -\sum p(x)\log q(x) \tag{7}$$

$p(x)$代表真实概率分布，$q(x)$代表预测概率分布，交叉熵损失是对两个概率分布差距的评估。交叉熵的函数图像（以Sigmod函数作为激活函数）（图6），可以看到交叉熵图像具有单调性并且损失越大梯度越大，因此训练时权重可以很好地进行更新。

3.2 自洽检验预测效果

自洽检验的目的是为了证明模型的合理性，由于样本不均衡可能带来收敛难度增大等问题，数量过多的样本可以被人为地随机丢弃一部分，使要进行分类的各种序列数量保持一致（即达到样本均衡），并且将分布均匀的样本作为训练集，测试对

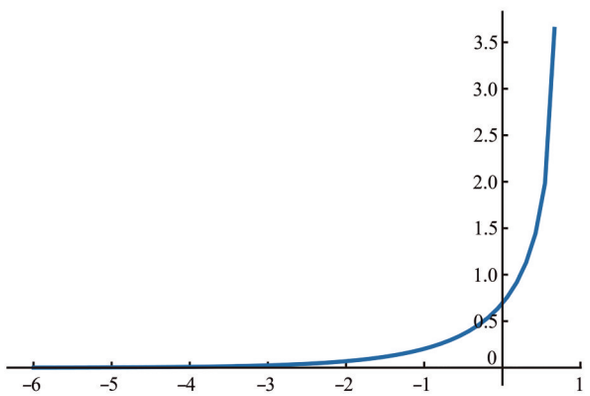

Fig. 6　Cross_Entropy curve

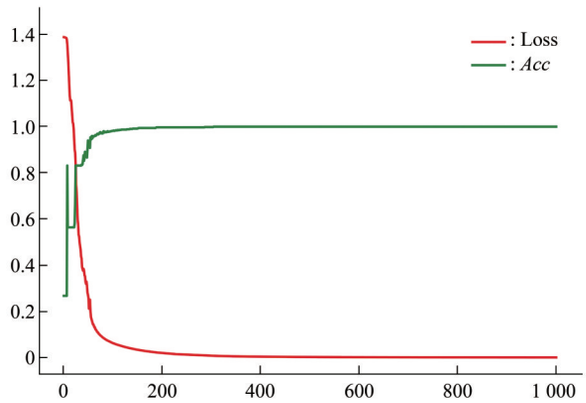

Fig. 7　Self−consistent verification training curve

全部样本的预测能力，以此来检验均衡样本对总体样本的预测能力（图7）。

对样本参数进行评估，可以看到整体的预测性能比较好，各项参数都有比较好的表现（表2）。

Table 2　Self−consistent verification overall evaluation Parameters

	F1-core (macro/micro)	Acc	Kappa	Hamming	Sample distribution
Coding	0.969 1/0.984	0.984	0.973 3	0.016	Evenly distributed
Non-coding	0.965 6/0.978 7	0.979	0.963 7	0.021	

图8为两种预测模型的ROC曲线，每个种类模型的AUC值均在0.96以上，模型表现出了良好的分类能力。

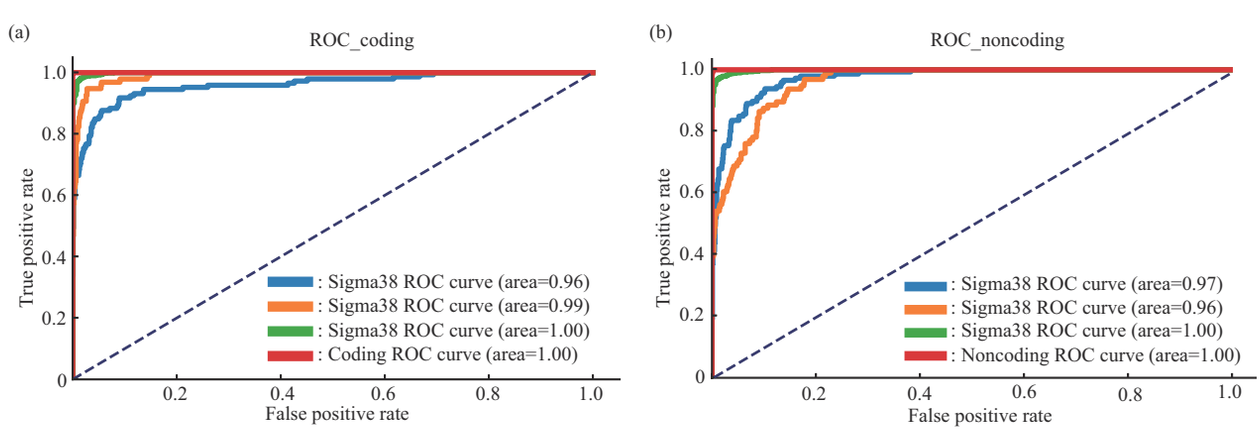

Fig. 8　Self−consistent verification of two four−category ROC curves

需要注意的是，均衡样本的优势在于可以快速收敛并且可以有效地防止过拟合现象的发生，它的训练次数较少并且学习率较高，使用的分类器较少，可以很好地减少内存损耗，预测精度也十分可观。人为制造均衡样本的缺点在于训练集可能不能完全包含样本的所有特征从而影响预测准确率，为此通过对量大的样本进行多次采样，然后生成多个分类器，最后对所有分类器结果求平均可以解决此问题。

3.3　独立检验

从图9中可以看到独立检验的训练取得了良好的效果，损失函数曲线以及准确率曲线都比较平

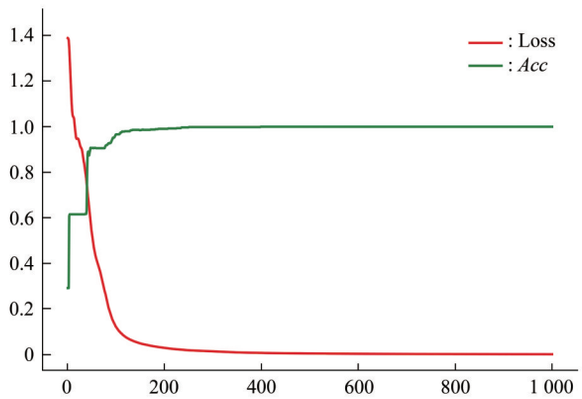

Fig. 9　Independent inspection training curve

滑，这说明训练的过程比较顺利。可以看到准确率曲线最终上升到了1.0，损失函数下降到了一个较低的位置（0.000 6）。

第999次的准确率为tensor（1）；第999次的损失函数为：tensor（0.000 6，grad_fn=<NegBackward>）。

下面的表中展示了独立检验的验证结果（平均结果），With-coding表示Sigma和Coding序列的验证结果（表3），With-Noncoding表示Sigma和Noncoding序列的验证结果（表4）。

Table 3　Independent inspection result（With-coding）

Dataset	Promoter	*Acc*	S_n	S_p	AUC
With-Coding	Sigma38	0.931 5	0.931 5	0.998 4	0.986 0
	Sigma54	1.0	1.0	0.991 0	0.999 5
	Sigma70	0.997 5	0.997 5	0.984 0	0.999 8
	Coding	1.0	1.0	0.988 6	1.0

Table 4　Independent inspection result（With-noncoding）

Dataset	Promoter	*Acc*	S_n	S_p	AUC
With-noncoding	Sigma38	0.931 5	0.931 5	1.0	0.991 6
	Sigma54	1.0	1.0	0.992 1	0.998 9
	Sigma70	1.0	1.0	0.981 9	0.999 9
	Non-coding	1.0	1.0	0.990 5	1.0

表3、4显示Sigma38序列相对于其他启动子的准确率明显较低，这可能是由于过拟合导致的，在训练中Sigma70也较容易出现这种情况。

独立检验绘出的ROC曲线（图10、11），可以看到无论是哪种样本组合，曲线的表现都十分让人满意。相对地来说，Sigma38的结果较差一些，这也是和上面的*Acc*结果有很好的对应。

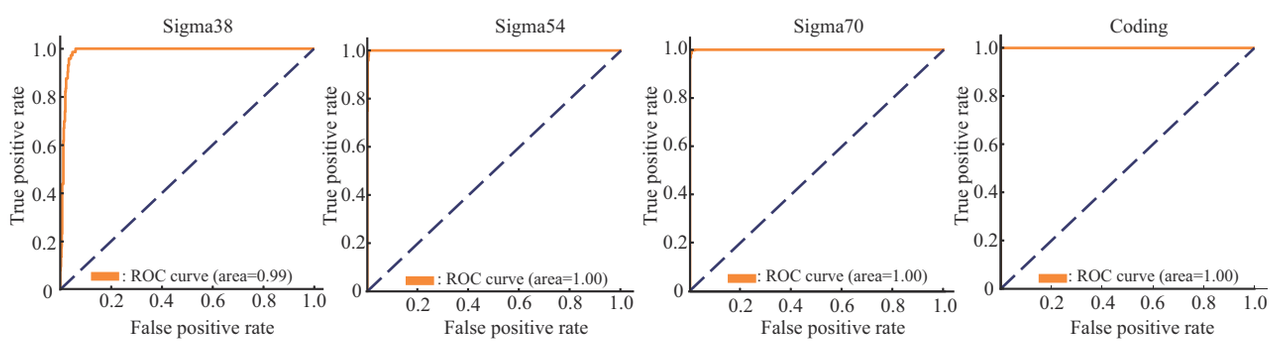

Fig. 10　Independent inspection ROC curve (With-coding)

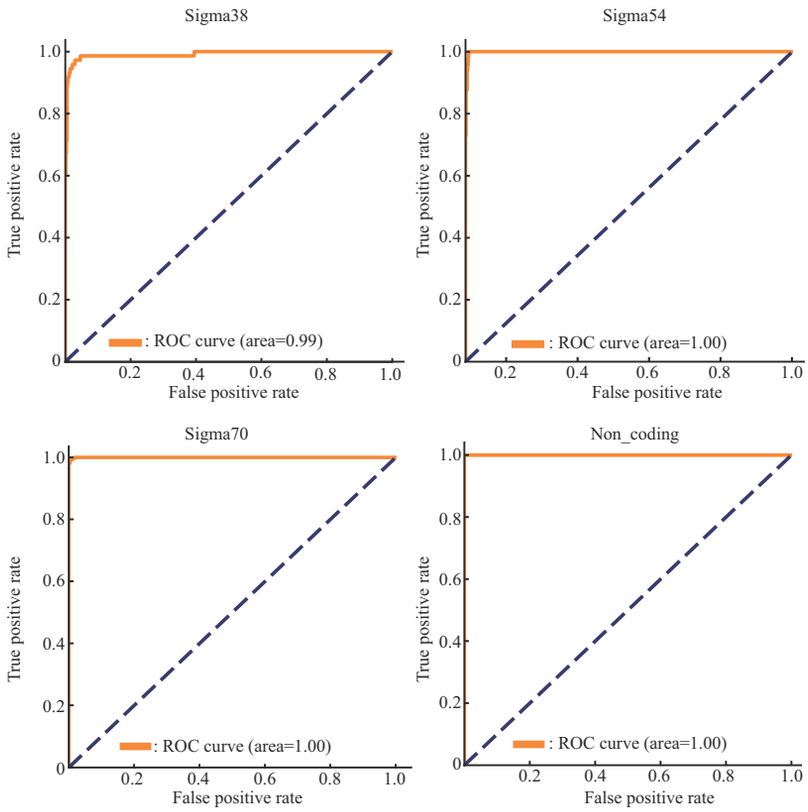

Fig. 11　Independent inspection ROC curve (With−noncoding)

独立检验的结果证明（表5），训练非均衡样本依然是可行的途径，这主要归功于CNN分类算法的高精度和PSSM能够较好地代表序列的特征。但是非均衡样本会带来收敛过慢甚至出现无法收敛的灾难，因此常常需要对其进行过采样处理（采用较低的学习率和较多的训练次数、更多的分类器以提取更多特征，如本次卷积层使用了16×16甚至是20×20的卷积核，而均衡样本使用的卷积核为5×5），这样就又增大了过拟合的风险，因此训练成功的难度也相较于均衡样本高，并且它对算力的损耗也增大了。

Table 5　Independent inspection overall evaluation parameters

	F1-core (macro/micro)	*Acc*	Kappa	Hamming	Sample distribution
Coding	0.983 1/0.991 6	0.991 6	0.985 9	0.008 4	Unevenly distributed
Non-coding	0.985 1/0.992 6	0.992 7	0.987 2	0.007 3	

Coding represents the four classifiers of promoters and coding, Non-coding refers to the four classifiers of promoters and noncoding.

3.4　十交叉检验

为了节省算力开销，实验结论通过十交叉检验法被验证（表6~8）。

Table 6　Ten−fold inspection result (With−coding)

Dataset	Promoter	*Acc*	S_n	S_p	AUC
With-coding	Sigma38	0.957 8	0.957 8	0.983	0.97
	Sigma54	1.0	0.99	0.975 9	0.999 6
	Sigma70	0.978 9	0.978 9	0.972 2	0.96
	Coding	0.957 8	0.957 8	0.983	0.990 9

Table 7　Ten-fold inspection result (With-noncoding)

Dataset	Promoter	*Acc*	S_n	S_p	AUC
With-noncoding	Sigma38	1.0	1.0	0.976 3	0.98
	Sigma54	0.99	0.99	0.979 6	0.99
	Sigma70	0.95	0.95	0.996 7	0.95
	Non-coding	1.0	1.0	0.976 3	0.99

Table 8　Ten-fold inspection overall evaluation parameters

	F1-core（macro/micro）	*Acc*	Kappa	Hamming	Sample distribution
Coding	0.976 7/0.976 6	0.976 7	0.968 9	0.023 3	Evenly distributed
Non-coding	0.982 3/0.982 2	0.982 2	0.976 3	0.017 8	

绘出的ROC曲线如下：

启动子和Coding区序列四分类的ROC曲线（图12）均在对角线上方，并且AUC值均比较接近1（Sigma38的AUC值为0.97，Sigma54和Coding序列的AUC均为0.99，Sigma70为0.96），这表明模型对于启动子的分类效果比较理想。

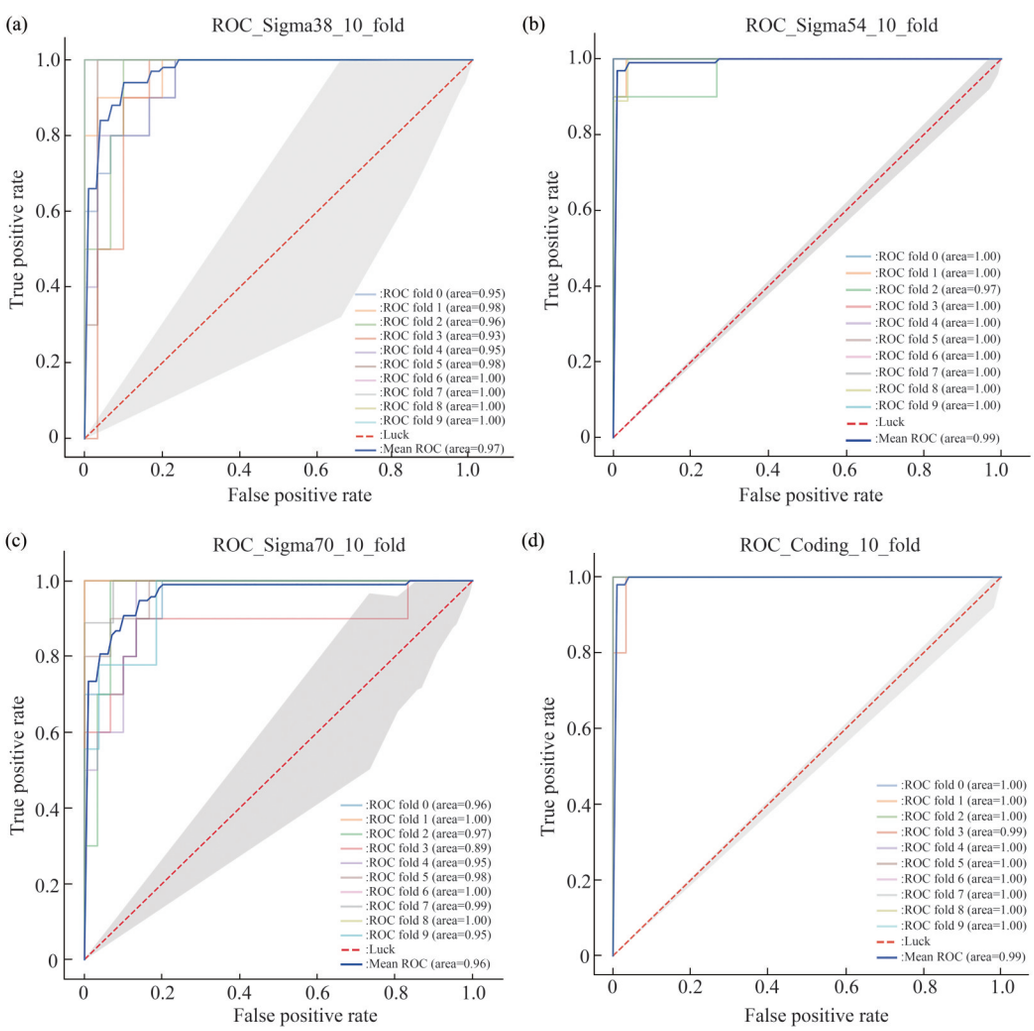

Fig. 12　Ten-fold inspection (With_coding) ROC curve

在启动子和Non_coding区序列四分类的ROC曲线（图13）中，Sigma54序列和Non_coding区序列依旧达到了0.99，Sigma38的AUC值上升到了0.98。

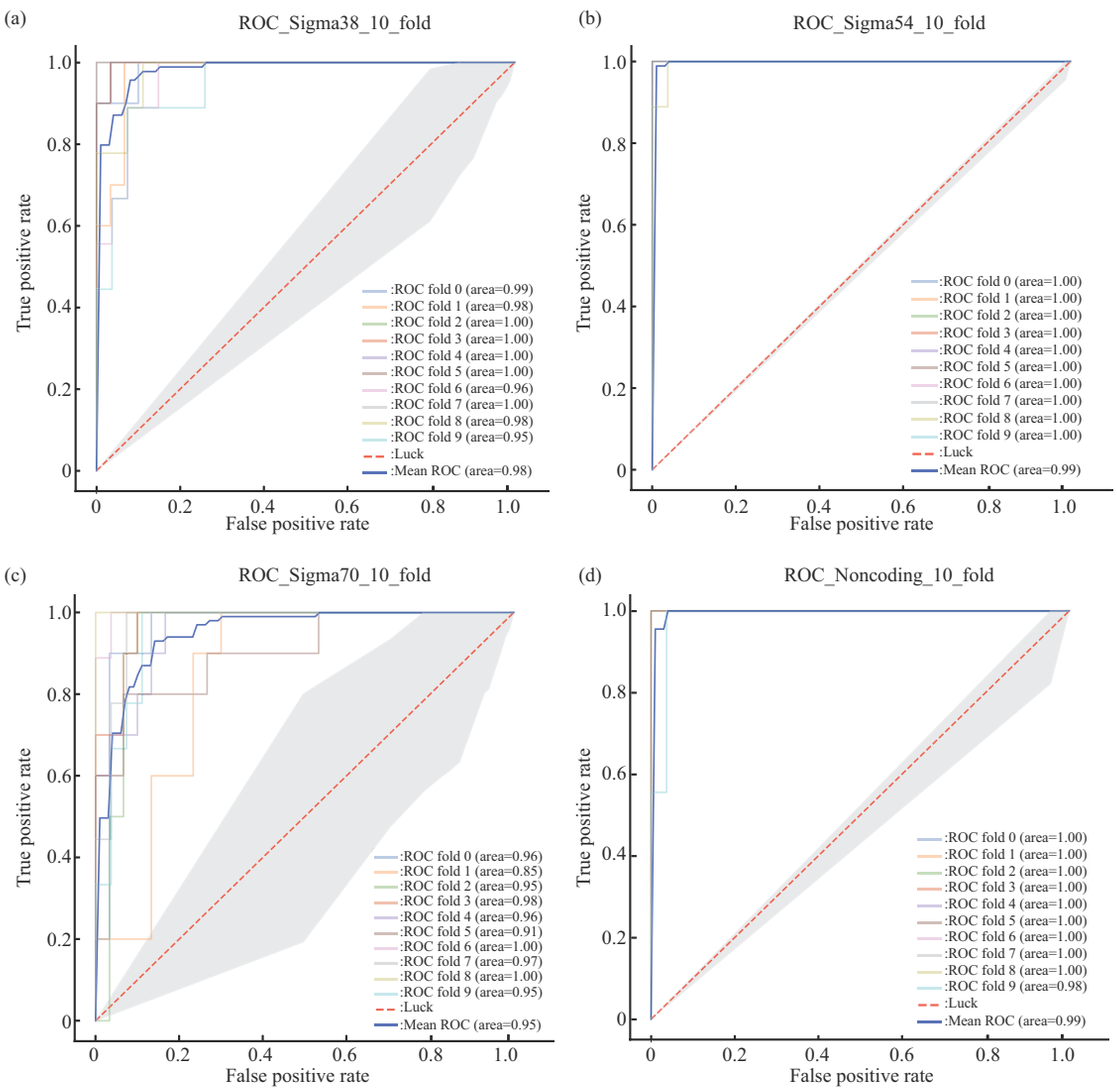

Fig. 13　Ten-fold inspection (With_noncoding) ROC curve

在十交叉检验的结果中，可以看到模型对于四分类的整体预测准确性都达到了0.97以上，并且对每一种序列的预测精确性也都达到了0.95以上。

3.5　对比分析

采用PSSM特征和采用二联体+柔性参数[11]两种方法得到的准确性（表9）结果的对比数据显示：PSSM特征的预测效果更为理想，并且采用本论文中方法取得的效果远远好于二联体+柔性参数的方法，这说明PSSM特征对于序列特征的描述更为精确。

单独使用PSSM特征分类和采用PSSM特征+CNN算法分类两种方式对Sigma38、Sigma54和Sigma70的预测结果（表9）的对比数据显示：CNN算法对3种启动子的预测准确性都优于仅仅使用PSSM特征分类的方式，而且CNN算法对每种启动子的预测准确性都比较均衡，没有出现对某个启动子预测精度过小的现象.

为了更进一步探究算法对启动子和非启动子的分类效果，两种方法（表10）对相同的数据集预测并且进行了十交叉验证，对比结果显示：本论文中的方法取得了更理想的结果。本论文方法在不同分类条件下得出的结果（表9，10）对比显示：二

分类的效果要好于多分类。

本论文取得的成果与同行的最新研究成果对比显示（表11）：Grad-CAM编码方法（Feature by Grad-CAM）可以取得稍好的准确率，但其特异性尤其是AUC值表现较为逊色，说明其模型稳定性可能存在问题，本论文则兼顾了准确率和模型参数两方面。独热编码（one-hot encoding）方法取得的准确率为0.901，AUC值为0.957 2，本论文的结果相对较好[24-25]。

Table 9 Comparison of Acc（Ten-fold inspection）

	Sigma38	Sigma54	Sigma70
PSSM+CNN	0.978 9	0.995	0.964 4
Only PSSM	0.96	0.97	0.74
Single parameter+Doublet	0.86	0.83	0.86

Table 10 Comparison of comprehensive parameters（Ten-fold inspection）

	Acc	S_n	S_p	AUC
PSSM+CNN	1.0	1.0	1.0	0.99
PseKNC+PCSF feature +SVM	93.1	92.2	91.2	0.976

Table 11 Comparison of comprehensive parameters (with the newest results)

	Acc	S_p	AUC	Published
PSSM+CNN	0.98/ 0.995 /0.964	0.98/0.98/0.98	0.97/0.99/0.95	None
Feature by Grad-CAM	0.99/0.999/0.979	0.66/0.65/0.78	0.63/1.0/0.84	2020.6
PSSM+CNN	0.979	0.980 4	0.978 8	None
One-hot encoding	0.901	0.903 8	0.957 2	2020.12

0.98/0.995/0.964 are parameters of Sigma38、Sigma54、Sigma70 respectively, 0.979 is the overall evaluation parameter.

4 讨 论

CNN+PSSM方法采用的特征简单易用，并且多分类可以大幅提高预测效率。有研究者单独采用PSSM打分进行分类，这种方式取得的效果稍逊于本文方法，主要原因可能是PSSM特征稍显简单。在没有有效算法辅助的情况下，这种分类方式相对利用更复杂、覆盖差异更全面的特征描述方法表现确实逊色。但是这并不意味着利用简单的特征描述方法得不到好的结果。Shujaat等[24]和Zhang等[25]都利用较简单的特征描述（01序列为基础）得到了较好的预测结果（前者97%以上，后者80%左右）。其次是在分类算法上，Shujaat等[24]采用了过于简单的01序列造成了很大的泛化，本身01序列蕴含信息就比较少，再次提取特征只会让CNN模型的过拟合风险增大。Zhang等[25]虽然取得了较好的准确率，但是在AUC（仅在0.84以上，且Sigma38的AUC为0.63）和Sp（0.78以下）等模型评估参数的表现上不如人意，其采用的序列转换方法（由字符序列转换为数字序列的方法）也是01编码方式。

样本不均衡容易造成训练的泛化，目前还没有好的调参办法来完美解决这一问题，并且这一点也常被同行研究人员所忽视。目前大多数深度学习的研究者采用的方法是减少训练样本中数量过多的种类的数量，人为地调整样本分布，或者采用过采样的方式对数量较少的种类重复取样。有研究者提出过采样可能是在以ROC/AUC作为评价指标时最佳

的处理方式[26-27]。为了减少模型拟合困难问题的发生，本文采用了人为调整样本数量的方法，剔除了过多的样本。

CNN和PSSM特征结合是采取了两条腿走路的方法：选取良好的特征描述方法可以最大限度的覆盖不同种类启动子之间的差异，为CNN提取差异进而进行有效的分类奠定基础；CNN自学习的优良特性和对算法的优化也是提升模型性能的关键。由于PSSM特征对启动子序列描述较为全面，因此本方法在序列转换过程中丢失了更少的信息，同理如果可以选择特征描述更为全面的转换方法，模型的准确率会有进一步的提升。

5 结　论

本论文通过构建样本的位点特异性打分矩阵，并且使用样本的位点特异性打分矩阵将待预测的字符序列转化成数值序列。利用PSSM特征训练出来的CNN模型对Sigma38、Sigma54和Sigam70 3种序列进行预测，分别得到了0.978 9、0.995、0.964 4的预测精确度。

在将序列数值化的过程中，PSSM特征能够很好地表征每种序列的核苷酸分布信息，这使得每种序列之间的区分度比较明显。由于PSSM特征的构建方法简单，因此该特征对于序列的特征表述不会过于冗杂，这有效地降低了CNN在训练模型时发生过拟合的风险。本文提出了一种解释序列特征的新思路，即利用类图片特征构建序列的点阵图像，这为下一步研究序列特征提取提供了一个新的方向：例如，如果可以基于PSSM再创造一套多通道的标准（类似于RGB标准），让每个位点的数值由多个通道共同决定，那么将序列展开为多维矩阵的识别效果可能更好[28]。

参　考　文　献

[1] Roy A L, Singer D S. Core promoters in transcription: old problem, new insights. Trend Biochem Sci, 2015, **40**(3): 165-171

[2] Thöny B, Hennecke H. The -24/-12 promoter comes of age. FEMS Microbiol Rev, 1989, **5**(4): 341-357

[3] 张颖, 贾芸, 吕军. 大肠杆菌σ^{70}启动子的识别. 生物物理学报, 2007, **23**(6): 475-481
Zhang Y, Jia Y, Lv J. The recognition of σ^{70} promoters in escherichia coli k-12. Acta Biophysica Sinica, 2007, **23**(6): 475-481

[4] 郭东华. 基于序列和结构信息识别大肠杆菌与人类启动子[D]. 呼和浩特:内蒙古大学物理科学与技术学院, 2020
Guo D H. Identifying *E. coli* and Human Promoter Based on Sequence Information and Structure Information[D]. Hohhot: School of Physical Science and Technology, Inner Mongolia University, 2020

[5] 左永春. 基于多类特征融合的基因启动子相关问题的理论研究[D]. 呼和浩特:内蒙古大学物理科学与技术学院, 2011
Zuo Y C. The Theoretical Studies of Promoter Based on Multi-features Fusion[D]. Hohhot: School of Physical Science and Technology, Inner Mongolia University, 2011

[6] 韩玲. 基于模体识别和机器学习的细菌基因组中sigma-54启动子预测 [D]. 济南:山东大学数学学院, 2018
Han L. Computational Prediction of Sigma-54 Promoters in Bacterial Genomes by Integrating Motif Finding and Machine Learning Stategies [D]. Jinan: School of Mathematics, Shandong University, 2018

[7] 周晖杰, 史定华. 基于知识神经网络的大肠杆菌启动子识别算法的改进及C++实现. 应用数学与计算数学学报, 2004, **18**(2): 1-7
Zhou H J, Shi D H. Communication on Applied Mathematics and Computation, 2004,**18**(2): 1-7

[8] 林昊, 李前忠. 大肠杆菌sigma70启动子预测. 生物信息学, 2007, **5**(1): 12-14
Lin H, Li Q Z. Chinese Journal of Bioinformatics, 2007, **5**(1): 12-14

[9] 丁辉, 邓恩泽, 陈伟, 等. 细菌σ^{54}启动子序列分析与预测. 电子科技大学学报, 2015, **44**(1): 147-149
Ding H, Deng E Z, Chen W, *et al*. Journal of University of Electronic Science and Technology of China, 2015, **44**(1): 147-149

[10] 闫妍, 万平. 利用位点特异性打分矩阵对大肠杆菌启动子的预测. 生物信息学, 2015, **13**(2): 125-130
Yan Y, Wan P. Chinese Journal of Bioinformatics, 2015, **13**(2): 125-130

[11] 谢亚茹, 董志飞, 杨佳赫, 等. 基于序列柔性参数的大肠杆菌启动子的预测. 内蒙古大学学报(自然科学版), 2018, **49**(6): 620-628
Xie Y R, Dong Z F, Yang J H, *et al*. Journal of Inner Mongolia University (Natural Science Edition), 2018, **49**(6): 620-628

[12] 赖洪燕. 基于序列顺序与位置信息的启动子预测[D]. 成都:电子科技大学, 2018
Lai H Y. Based on Sequence-order and Position-correlation Information Recognizing Promoters[D]. Chengdu: University of Electronic Science and Technology of China, 2018

[13] 胡宇佳, 甘伟, 朱敏. 基于多特征融合的增强子-启动子相互作用预测综述. 计算机科学, 2020, **47**(5): 64-71
Hu Y J, Gan W, Zhu M. Computer Science, 2020, **47**(5): 64-71

[14] Ayub S I, Rozaimi M R, Salim B, *et al*. bTSSfinder: a novel tool for the prediction of promoters in cyanobacteria and *Escherichia coli*. Bioinformatics (Oxford, England), 2017, **33**(3): 334-340

[15] 梁志勇. 启动子预测算法研究与软件开发[D]. 成都:电子科技大学, 2017
Liang Z Y. The Research and Software Development on Prediction Algorithm of Promoter[D]. Chengdu: University of Electronic Science and Technology of China, 2017

[16] Anwar F, Baker S M, Jabid T, *et al*. Pol II promoter prediction using characteristic 4-mer motifs: a machine learning approach. BMC Bioinformatics, 2008, **9**: 414

[17] Uwe O. Identification of core promoter modules in Drosophila and their application in accurate transcription start site prediction. Nucleic Acids Res, 2006, **34**(20): 5943-5950

[18] Burden S, Lin Y X, Zhang R. Improving promoter prediction Improving promoter prediction for the NNPP2.2 algorithm: a case study using *Escherichia coli* DNA sequences. Bioinformatics, 2005, **21**(5): 601-607

[19] Lu J, Luo L F . Prediction for human transcription start site using diversity measure with quadratic discriminant. Bioinformation, 2008, **2**(7): 316-321

[20] Umarov R K, Solovyev V V. Recognition of prokaryotic and eukaryotic promoters using convolutional deep learning neural networks. PLoS One, 2017, **12**(2): e0171410

[21] 张淑军, 张群, 李辉. 基于深度学习的手语识别综述. 电子与信息学报, 2020, **42**(4): 1021-1032
Zhang S J, Zhang Q, Li H. Journal of Electronics & Information Technology, 2020, **42**(4): 1021-1032

[22] 时梨, 蔡林. 基于Python语言构建神经网络识别手写数字的研究. 电脑编程技巧与维护, 2021, **428**(2): 117-118+130
Li S, Cai L. Computer Programming Skills & Maintenance, 2021, **428**(2): 117-118+130

[23] Lakhane P, Rane M. Handwritten digital recognition. J Res Sci Eng, 2020, **2**(12): 89-92

[24] Shujaat M, Wahab A, Tayara H, *et al*. pcPromoter-CNN: a CNN-based prediction and classification of promoters. Genes, 2020, **11**(12): 1529

[25] Zhang M, Wang L, Wan P. Discovering *Escherichia coli* K-12 promoter features using convolutional neural network. Comput Biol Bioinform, 2020, **8**(1): 15-19

[26] Buda M, Maki A, Mazurowski M A. A systematic study of the class imbalance problem in convolutional neural networks. Neural Networks, 2018, **106**: 249-259

[27] Abbas M M, Mohie-Eldin M M, El-Manzalawy Y. Assessing the effects of data selection and representation on the development of reliable *E. coli* sigma 70 promoter region predictors. PLoS One, 2015, **10**(3): e0119721

[28] Long W , Li T G , Yang Y, *et al*. FlyIT: *Drosophila* embryogenesis image annotation based on image tiling and convolutional neural networks. IEEE/ACM Trans Comput Biol Bioinform, 2021, **18**(1): 194-204

Prediction of *E.coli* Promoters Based on CNN*

PENG Bao-Cheng[1)], ZHANG Xiao-Wei[2)**], LIU Yang[2)], Fan Guo-Liang[1)**]

([1)]*School of Physical Science and Technology, Inner Mongolia University, Hohhot* 010021, China;

[2)]*Department of Rheumatology, the First Affiliated Hospital, Inner Mongolia Medical University, Hohhot* 010050, China)

Abstract **Objective** The prediction model based on PSSM (position-specific scoring matrix) has achieved good results, and various optimization methods based on PSSM are also being continuously developed. However, the accuracy rate is relatively lower. In order to further improve the prediction accuracy rate, this paper does further research based on the CNN algorithm. **Methods** In this paper, PSSM is used to process the letter sequence into a numeric matrix, and through a convolutional neural network (CNN) algorithm for classification. The 3 promoter sequences of Sigma38, Sigma54 and Sigma70 of *E.coli* K-12 (*Escherichia coli* K-12, hereinafter referred to as *Escherichia coli*) are used as the positive sets, and the sequences of the Coding and Non-coding regions of *Escherichia coli* are the negative set. **Results** In the prediction of *Escherichia coli* for the two-classification for promoters, the accuracy rate reaches 99%, and the success rate of promoter prediction is close to 100%; in the three-classification for Sigma38, Sigma54 and Sigma70 promoters, the prediction accuracy rate is 98%, and for each the prediction accuracy of these sequences can reach 0.98 or more. Finally, we tried 4 classifications of 3 promoters of Sigma38, Sigma54 and Sigma70 with Coding area or Non-coding area sequences respectively, the accuracy of prediction was 0.98. The prediction accuracy of the ten-fold cross-validation of the balanced samples of the Sigma promoters can reach more than 0.95, the Hamming distance is 0.016, and the Kappa coefficient is 0.97. **Conclusion** Compared with other classification algorithms such as SVM (support vector machine), the CNN classification algorithm has more advantages, and based on the classification advantages of CNN, the coding method can also be simplified.

Key words *Escherichia coli*, position specific scoring matrix, CNN, multi-classification

DOI：10.16476/j.pibb.2021.0139

* This work was supported by grants from The National Natural Science Foundation of China (62063024), The Scientific Research Program at Universities of Inner Mongolia Autonomous Region of China (NJZY20005) and The Students Innovation Training Program of the Inner Mongolia University (201912240).

** Corresponding author.

FAN Guo-Liang Tel：86-471-4992958，E-mail：fanguoliang@imu.edu.cn

ZHANG Xiao-Wei Tel：86-13948518183，E-mail：13948518183@126.com

Received：May 13，2021　Accepted：October 21，2021