International Journal of Applied Science and Engineering
2004. 2, 2: 177-188

Prediction of RNA Polymerase Binding Sites Using Purine-Pyrimidine Encoding and Hybrid Learning Methods

Cheng-Jian Lin [a*], Chun-Cheng Peng [b], and Chi-Yung Lee [c]

[a] *Department of Computer Science and Information Engineering,*
Chaoyang University of Technology,
Wufeng, Taichung County 413, Taiwan, R.O.C.
[b] *School of Computer Science and Information Systems,*
Birkbeck, University of London,
London WC1E 7HX, UK
[c] *Department of Electronic Engineering,*
Nan Kai College,
Caotun, Nantou County 542, Taiwan, R.O.C.

Abstract: *Escherichia coli* (*E. coli*) K12 was sequenced in 1997. The 4,639,221-base pair DNA sequence consists of 4288 annotated protein-coding genes, 38 percent of which have no attributed function. One of the major problems in predicting prokaryotic promoters is locating the spacers between the -35 box and -10 box and between the -10 box and transcription start site. In this paper, we use the adopted expectation maximization (EM) algorithm to accurately find the localizations of the promoter regions. A brand new purine-pyrimidine encoding method is proposed to reduce the dimensions of the training data. The heavy demand on systems for both computation and memory space can then be avoided through the choice of coding factor. The most representative features are used for training learning vector quantization networks. The simulation results of the proposed coding approach reveal that the precision of promoter prediction using the proposed approach is approximately the same as the precision using the traditional encoding method.

Keywords: *E. coli*; promoter prediction; purine-pyrimidine; expectation maximization algorithm; learning vector quantization networks.

1. Introduction

E. coli has been studied for over one hundred years since its successful isolation in 1885. Being the model microorganism, *E. coli* has become a ‘living platform’ for many biological and chemistry experiments, such as vectors transcription, plasmid cloning, medical drug design, etc. But many unsolved problems still exist for the organism, i.e., how and and when the *E. coli* genes decide to express the given functionalities and where and what subparts in the genome respond to these regulating tasks.

With the progress of molecular biology and modern DNA sequencing techniques, more than 194 organisms have been sequenced and annotated [1]. The first substrain of *E. coli*, K12, was sequenced in 1997. The 4,639,221 base pair (bp) DNA sequence consists of 4288

[*] Corresponding author; e-mail: cjlin@mail.cyut.edu.tw

Accepted for Publication: June 16, 2004

annotated protein-coding genes, 38 percent of which have no attributed function [2]. Promoters, being transcriptional signals and lying in the RNA polymerase contact region, regulate gene expressions. In the annotated data, many promoter regions have not yet been determined. Studies of four additional sequencing tasks for *E. coli* strains (and substrains) have been completed most recently. But many vague promoter regions still need to be explored.

Characterization and recognition of promoter regions are important research topics and have been studied by many researchers. The *E. coli* promoter is located immediately before the *E. coli* gene. Thus, successfully locating the *E. coli* promoter leads to identifying the *E. coli* gene. The uncertain characteristics of the *E. coli* promoters contribute to the difficulty of recognizing and predicting promoters [3].

Each *E. coli* promoter contains two binding sites to which the *E. coli* RNA polymerase, a kind of protein, binds. The two binding sites are the minus35 (35 nucleotides upstream of the transcriptional start site) region and the minus10 (10 nucleotides upstream of the transcriptional start site) region. (The transcriptional start site is the first nucleotide of a codon where the transcription begins; it serves as a reference point.) Both of the two binding regions are hexamer boxes. The consensus sequences, that is, the prototype sequences composed of the most frequently occurring nucleotides at each position, the minus 35 binding site and the minus 10 binding site, are **TTGACA** and **TATAAT**, respectively. However, few existing *E. coli* promoters exactly contain these two consensus sequences.

Many researchers have used artificial intelligence approaches to improve the learning abilities for generalization purposes [4-7]. The unique computing architectures that neural networks potentially provide have attracted interest from researchers across different disciplines. As a technique for computational analysis, neural network technology is very well suited for the analysis of molecular sequence data [8]. The perceptron algorithm was revised for bio-sequence analysis in an attempt to distinguish DNA/RNA ribosomal binding sites from non-binding sites [9]. The backpropagation algorithm also has been successfully used to perform a variety of input-output mapping tasks for recognition, generalization, and classification [10], as well as many molecular sequence analysis problems. Most early sequence analysis studies involved the use of perceptron or backpropagation networks for protein structure prediction or DNA sequence discrimination [8]. As this field continues to develop, researchers have broadened the choices of neural network architectures and have learned paradigms to solve a wider range of problems.

With the progress of modern sequencing technologies, more and more sequencing tasks can be finished. Thus, more and more annotation data need to be experimentally verified and computationally predicted. We also found that almost all related studies take no more than 500 promoter patterns to train their prediction systems. In [11], we chose three new compilations of *E. coli* K12 promoter prediction researches as our positive training data sets. These three training sets were 362, 441, and 421 positive patterns with 65, 80, and over 300 bps long, respectively. But a major problem still existed: most of these patterns did not indicate where the promoter regions were. In addition, the computational process was extremely long. To overcome these problems, we adopted the expectation maximization (EM) algorithm to locate and learn the distribution of these positive promoter regions. Then we applied our new encoding method, the purine-pyrimidine approach, to reduce the input dimensions. Preliminary analysis has shown that our encoding method is lightly better than other coding approaches. All these coded patterns are fed into the neural networks to verify the precision of predictions.

This paper is organized as follows. Section 2 discusses the biological data of *E. coli* and the characteristics of prokaryotic promoters. In Section 3, we describe the spacer locating EM algorithm and the prediction of EM extracted sequence via different neural networks. Section 4 describes the prediction of our purine-pyrimidine encoded sequence. The benefits of our brand new encoding method and the comparison between traditional coding approach and our approach are also presented. The conclusion and plans for future works are provided in the last section, Section 5.

2. Characteristics of prokaryotic promoters

Prokaryotic promoters appear to be less complex (the size and number of elements are recognizable by sigma factors) than their eukaryotic counterparts. There are some similarities, though. For example, both are recognized by other factors before RNA polymerase binding.

Prokaryotic promoters vary in their affinities for RNA polymerase, a factor very important with regards to controlling the frequency of transcription and, therefore, the extent of gene expression. Unregulated transcription initiation at many prokaryotic promoters has been found to require only an RNA polymerase holoenzyme, which consists of four core subunits with a dissociable sigmạ factor. Multiple sigmạ factors have been identified, and each sigma factor programs the core enzyme for transcription from a different class of promoters.

Prokaryotic promoters direct not only the site of transcription initiation but also the rate of transcription. Earlier studies [12, 13] have established that promoter strength, as defined by the degree in which transcripts of the corresponding genes are produced, is primarily determined by two factors: the binding affinity to RNA polymerase and the rate of isomerization from 'closed promoter complexes' (DNA remains duplex) to 'open promoter complexes' (DNA opened by 'melting').

There are four notable features in most *E. coli* promoters: the transcriptional start site, the -10 hexamer, the -35 hexamer, and the distance between the -10 and -35 sequences (see Figure 1). The transcriptional start site has been found to be purine in more than 90% of characterized promoters [14]. It is common for the transcription start site to be the central base within the sequence **CAT**, but the conservation of this triplet is not great enough to regard it as an obligatory signal. Just upstream of the start site, a six base pair (bp) region is recognizable in most promoters. The center of the hexamer is often close to 10 bp upstream of the TSS. The distance in known promoters varies from 18 to 9 from the transcriptional start site. In some other literature, this range varies from 11 to 3 [3]. Named for its location, the hexamer is often called -10 box. Its consensus is **TATAAT** and can be summarized in the form $\mathbf{T}_{80}\mathbf{A}_{95}\mathbf{T}_{45}\mathbf{A}_{60}\mathbf{A}_{50}\mathbf{T}_{96}$, where the subscripts denote the percentage of occurrence of the most frequently found base[1]. The canonical -35 (**TTGACA**) and –10 hexamers (**TATAAT**) are located at positions 15 to 21 and 39 to 44, respectively. The promoter data was obtained from [14], and the informational analysis used is a program called sequence logo [15].

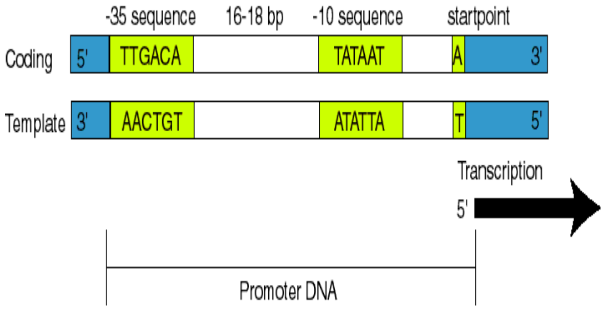

Figure 1. Typical prokaryotic promoter

The other conserved hexamer is around ~35 bp upstream of the start site. The consensus for -35 has been universally accepted as **TTGACA** [14]. In more detailed form, the conservation is $\mathbf{T}_{82}\mathbf{T}_{84}\mathbf{G}_{78}\mathbf{A}_{65}\mathbf{C}_{54}\mathbf{A}_{45}$ (Figure 1). The distance separating the -35 and -10 sites has been found to be between 16 and 18 bp in 90% of the promoters. With very unusual exceptions, it may be as short as 15 bp or as wide as 21 bp. The distance may be critical in holding the two sites at the appropriate distance for the geometry of a RNA polymerase. An ideal *E. coli* promoter may consist of the -35 hexamer separated by 17 bp from the -10 hexamer, with the -10 hexamer lying about 7 bp upstream of the start site. The -35 region is said to provide the signal for recognition by a RNA polymerase, while the -10 sequence allows the complex to convert from a 'closed' to an 'open' form [13].

Other researchers have established another important sequence element in some *E. coli* promoters in addition to the four mentioned [16, 17]. The seven *E. coli rrn* genes, which encode ribosomal RNA, are unusually strong, accounting for more than 60% of the total RNA system in rapidly growing cells. The exceptional strength of the *rrn* promoter has been attributed to an **AT**-rich sequence of ~20 bp located immediately upstream of the -35 region. This region with the **AT**-rich motif has been termed the upstream element or the UP element [17].

The authors used two pieces of evidence to establish that the UP element is recognized by a RNA polymerase. First, the UP element was found to function in vitro in a transcription system containing only purified RNA polymerase and the promoter DNA sequences. The second evidence was in DNAase I footprinting experiments, where a RNA polymerase was found to protect the UP element, yielding a ~20 bp extended footprint [18]. The UP element is believed to be functional as the face of the helix phasing is maintained with respect to the transcriptional start site. The functional nature of the UP elements when kept in phase with the helix was confirmed when mutations that change the spacer length in promoters altered the level of transcription in vitro [17].

3. Promoter prediction of extracted sequence

The data set of positive promoters used in this section is taken from the compilation result in [19]. The negative data set is randomly generated, with 60% **AT** composition, that is, 60% of the nucleotides of each pattern are adenine (**A**) or thymine (**T**). For cross validation reasons, all duplicates of the positive patterns were removed. Then we fed both positive and negative training examples into the adopted EM algorithm and extracted the features while the convergence condition was achieved. A purine-pyrimidine encoding method was developed to encode the training patterns. Then the encoded training patterns were used to train the learning vector quantization networks.

3.1. The spacers locating via EM algorithm

The EM algorithm [20] is a general method for finding the maximum-likelihood estimate of the parameters of an underlying distribution from a given data set when the data is incomplete or has missing values. There are two main applications of the EM algorithm. The algorithm can be used when the data has missing values due to problems with or limitations of the observation process. The algorithm can also be used when optimizing the likelihood function is analytically intractable but when the likelihood function can be simplified by assuming the existence of and values for additional but *missing* (or *hidden*) parameters. The latter application is more commonly used for computational pattern recognition.

Let T represent the set of training *E. coli* promoters, that is, T contains all positive training sequences. Let K denote the cardinality of T. For a promoter sequence $S_i \in T$, the length of the spacer between the minus 10 region and the transcriptional start site, denoted by *sp*$_{10}$, and the length of the spacer between the minus 35 region and the minus 10 region, denoted by *sp*$_{35}$, are unobserved, though S_i is observed. We refer to the positive training sequences as "observed" data since they are given. These observed data are incomplete because the lengths of the two spacers are not given. (These lengths are referred to as "unobserved" or "missing" data.)

In general, *sp*$_{10}$ varies from 3 to 11 and *sp*$_{35}$ varies from 15 to 21. For each S_i, the missing data *sp*$_{10}$ and *sp*$_{35}$ are represented by a vector $z_i = (z_{i,1}, \cdots, z_{i,63})$, where

$$z_{i,f(m,n)} = \begin{cases} 1, & \text{if } m = sp_{10}, \text{ and } n = sp_{35} \\ 0, & \text{otherwise} \end{cases} \qquad (12)$$

where $f(m,n) = (m-3)*7 + n - 15$ is used for indicating the spacers instance. Each binding site consists of six bases. Assume that the nucleotides at the two binding sites of a promoter sequence are independent. Let $P_{10,j}(x), j = 1, \cdots, 6$, denote the probability of x, $x \in D = \{$A, C, G, T$\}$, occurring at position j in the minus10 region, and let **P10** denote $(P_{10,1}, \cdots, P_{10,6})$. Also, let $P_{35,j}(x), j = 1, \cdots, 6$, denote the probability of x, $x \in D$, occurring at position j in the minus35 region, and let **P35** denote $(P_{35,1}, \cdots, P_{35,6})$. Thus, $P_{10,j}$ and $P_{35,j}$ are in the multinomial distribution. For each *E. coli* promoter sequence, if we know the lengths of the two spacers, we could easily calculate the model parameter θ.

The EM algorithm proceeds iteratively until convergence occurs. Every iteration consists of two steps: 1) an expectation step (E step) and 2) a maximization step (M step). The E step calculates the sum of the log of the prior probability of θ, Pr_θ, and the expected complete-data log likelihood, where the expectation is for the distribution of the missing data given the observed data and current estimates of θ. Thus, the E step calculates

$$E_{Z|T,\theta^T} \log P(T, Z \mid \theta) + \log Pr\theta \qquad (13)$$

Assume that all $S_i \in T$, $1 \leqq i \leqq K$, are independent, and $P(Z|\theta) = P(Z)$, that is, the probability distribution of unobserved data is independent of θ. Then

$$\begin{aligned} E_{Z|T,\theta^T} \log P(T, Z \mid \theta) &= E_{Z|T,\theta^T} \cdot \log P(T, Z \mid \theta) \cdot P(Z) \\ &= \sum_{i=1}^{K} \sum_{m=3}^{11} \sum_{n=15}^{21} P\left(z_{i,f(m,n)} = 1 \mid S_i, \theta^T\right) \cdot \log\left(P\left(S_i \mid z_{i,f(m,n)} = 1, \theta\right) \cdot P\left(z_{i,f(m,n)} = 1\right)\right) \end{aligned} \qquad (14)$$

Let $S_{i,j}$ denote the nucleotide at position j of the promoter sequence S_i. Define

$$I_{i,j,x} = \begin{cases} 1, & \text{if } S_{i,j} = x \\ 0, & \text{otherwise} \end{cases} \qquad (15)$$

For each S_i, given θ^t and $z_{i,f(m,n)} = 1$, the likelihood of S_i is

$$P\left(S_i \mid z_{i,f(m,n)}=1,\theta\right)=\prod_{j=1}^{6}P_{10,j}^{t}\left(S_{i,49-m+j}\right)\cdot\prod_{j=1}^{6}P_{35,j}^{t}\left(S_{i,43-m-n+j}\right) \tag{16}$$

From Baye’s law, we have

$$\begin{aligned}P\left(z_{i,f(m,n)}=1 \mid S_i,\theta\right)&=\frac{P\left(S_i \mid z_{i,f(m,n)}=1,\theta\right)\cdot P^t\left(z_{i,f(m,n)}=1\right)}{P\left(S_i \mid \theta^t\right)}\\&=\frac{P\left(S_i \mid z_{i,f(m,n)}=1,\theta\right)\cdot P^t\left(z_{i,f(m,n)}=1\right)}{\sum_{m=3}^{11}\sum_{n=15}^{21}P\left(S_i \mid z_{i,f(m,n)}=1,\theta^t\right)\cdot P^t\left(z_{i,f(m,n)}=1\right)}\end{aligned} \tag{17}$$

Leaving out the terms not involvingθ, we get the log of the prior of θ, Pr_θas follows:

$$\log\Pr_\theta=\sum_{j=1}^{6}\sum_{x=A}^{T}\left(\alpha_x^{10.j}-1\right)\cdot\log P_{10,j}\left(x\right)+\sum_{j=1}^{6}\sum_{x=A}^{T}\left(\alpha_x^{35.j}-1\right)\cdot\log P_{35,j}\left(x\right) \tag{18}$$

Then substituting Eq. (16) and Eq. (17) into Eq. (14), we have

$$\begin{aligned}&E_{Z|T,\theta^T}\log P\left(T,Z \mid \theta\right)+\log\Pr_\theta\\&=\sum_{j=1}^{6}\left(K+\alpha_x^{10,j}-4\right)\cdot\sum_{x=A}^{T}f_{10,j}\left(x\right)\cdot\log P_{10,j}\left(x\right)+\sum_{j=1}^{6}\left(K+\alpha_x^{35,j}-4\right)\cdot\sum_{x=A}^{T}f_{35,j}\left(x\right)\cdot\log P_{35,j}\left(x\right)\\&+K\sum_{m=3}^{11}\sum_{n=15}^{21}f_s\left(m,n\right)\cdot\log P\left(z_{i,f(m,n)}=1\right)\end{aligned} \tag{19}$$

where

$$\begin{aligned}f_{10,j}\left(x\right)&=\frac{1}{K+\alpha_x^{10,j}-4}\left(\alpha_x^{10,j}-1+\sum_{i=1}^{K}\sum_{m=3}^{11}\sum_{n=15}^{21}I_{i,49-m+j,x}\cdot P_{10,j}\left(z_{i,f(m,n)}=1 \mid S_i,\theta^t\right)\right)\\f_{35,j}\left(x\right)&=\frac{1}{K+\alpha_x^{35,j}-4}\left(\alpha_x^{35,j}-1+\sum_{i=1}^{K}\sum_{m=3}^{11}\sum_{n=15}^{21}I_{i,43-m-n+j,x}\cdot P_{35,j}\left(z_{i,f(m,n)}=1 \mid S_i,\theta^t\right)\right)\\f_s\left(m,n\right)&=\frac{1}{K}\sum_{i=1}^{K}\sum_{m=3}^{11}\sum_{n=15}^{21}P\left(z_{i,f(m,n)}=1 \mid S_i,\theta^t\right)\end{aligned} \tag{20}$$

Let θ^0 denote the value of θ at the beginning of the first iteration. θ was initialized to a random value so that the E step can proceed. In every iteration, we use the current estimateθ^t to calculate the sum of the log of the prior probability of θ and the expected complete data log likelihood. The M step maximizes Eq. (19) with respect to θ. According to the information theory [21], $\sum_{x=A}^{T} f_{10,1}(x)\log P_{10,1}(x)$ is maximized when $P_{10,1}(x)$ equals $f_{10,1}(x)$, where $f_{10,1}(x)$ is a constant. Thus, the MLE of θ includes samples $f_{10,j}$, $f_{35,j}$, and f_s, $j = 1,\cdots,6$. That is,

$$
\begin{aligned}
&P^{t+1}_{10,j}(x) = f_{10,j}(x), \quad x \in D\\
&P^{t+1}_{35,j}(x) = f_{35,j}(x), \quad x \in D\\
&P^{t+1}(z_{i,f(m,n)} = 1) = f_s(m,n), \quad x \in D
\end{aligned}
\tag{21}
$$

The new value of θ can be used in the next iteration. The process iterates to convergence. Given the model parameters calculated from the positive training sequences (i.e., the promoter sequences in the training dataset *T*), we can determine the locations of the two putative binding sites of any DNA sequence S_i, where S_i could be a positive or negative training sequence or an unlabeled test sequence, by choosing the two spacer lengths sp_{10} and sp_{35} that are calculated by

$$\max_{3\le m\le 11,\,15\le n\le 21}\left\{P(S_i, z_{i,f(m,n)} = 1)\middle|\theta\right\}.$$

This EM algorithm (see Figure 3) would be used for the related compilations of the *E. coli* promoter prediction. The extracted dataset consists of 35 bps DNA sequences (i.e., 17 bps in -35 region, 11 bps in -10 region, and 7 bps in +1 region) and 2 spacers (spacer 35 and spacer 10). We first apply the conventional encoding method to the DNA alphabet {**A**, **C**, **G**, **T**} as a 4-bit pattern of {1000, 0100, 0010, 0001}. If we assume that the extracted DNA patterns are *k* bps long, the encoded training data via orthogonal codes would have 4*k* dimensions each. Since the actual length of an *E. coli* promoter is still unknown to date, the value of *k* should always be set to a number that is larger than 33, the maximum length appearing in a sequenced annotation data file. It is an extremely large dimension for neural network computing.

3.2. The purine-pyrimidine encoding method

In order to reduce input dimensions, we propose a purine-pyrimidine encoding method. That is, more precisely speaking, any bit of extracted patterns is considered to belong to purine or pyrimidine. Then we look up a predefined codebook for the patterns to find the corresponding codes. Before this is done, the codebook should be first designed. Given a *k* bps long DNA pattern, the coding range *r* must be a factor of *k* and must satisfy the constraint $1 < r < k$. Another little trick in choosing the pattern length *k* is that *k* should not be a prime number.

The dimension of training data pairs can then be reduced from 4*k* (orthogonal encoding) to *k*/*r* (our approach). For example, assume that the coding range *r* in our approach is 2. Table 1 shows the predefined purine/pyramid ine codebook for *r*=2. Figure 4 shows the coded difference between these two methods. The DNA pattern for this example is the consensus sequence, **TATAAT**, of *E. coli* -10 box.

Table 1. Pseudo codebook for *r*=2

Patterns	**Purine** (A/G)	**Pyrimidine** (C/T)
Purine (A/G)	1	2
Pyrimidine (C/T)	3	4

```
Input: the positive training data set T, the negative training data set G,
and the test data set Q of DNA sequences.
Output:   the position weight matrices $\mathbf{P}_{10}$, $\mathbf{P}_{35}$,
  and the putative $sp_{10}$, $sp_{35}$ of each DNA sequences.

Initialize probability distribution $P_{10}^0$, $P_{35}^0$, and $P^0\left(z_{i,f(sp_{10},sp_{35})}\right)$;
Do{
  //the expectation step
  For each patterns $S_i \in T$
      For each possible value of $sp_{10}$ and $sp_{35}$
          Calculate $P(S_i \mid z_{i,f(m,n)} = 1, \theta^t)$ according to Eq. (16);
      For each possible value of $sp_{10}$ and $sp_{35}$
          Calculate $P(z_{i,f(m,n)} = 1 \mid S_i, \theta)$ according to Eq. (17);
  Calculate $f_{10,j}$, $f_{35,j}$, and $f_s$ according to Eq. (20);

  //the maximization step
  Calculate $P_{10}^{t+1}$, $P_{35}^{t+1}$, and $P^{t+1}\left(z_{i,f(sp_{10},sp_{35})}\right)$ according to Eq. (21);
} until the change of $P_{10}^{t+1}$, $P_{35}^{t+1}$, and $P^{t+1}\left(z_{i,f(sp_{10},sp_{35})}\right) \le$ a predefined threshold.
```

Figure 3. The EM algorithm for *E. coli* spacers locating

In Figure 4, the sample sequence, TATAAT, is in the second row and is surrounded by square boxes. The numbers in the boxes to which the arrows point indicate encoding results. The first row shows the coded result for the orthogonal method. Since one DNA base used four bits for coding, the orthogonal method needs 4*6=24 bits, i.e., 24 dimensions. The other rows show our purine-pyrimidine approach. For r=2 and r=3, our encoding method just produced 3-dimension and 2-dimension coded data, respectively. According to the predefined codebook, Table 1, we can encode TA (T for pyrimidine and A for purine) into the real number 3 and encode AT into 2. When k increases, the reduction of coded dimensions is more remarkable.

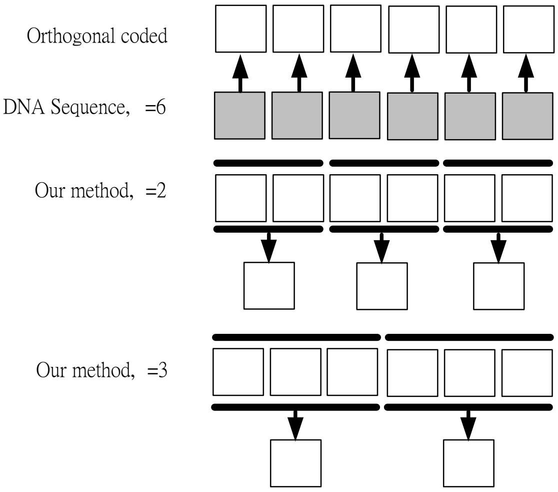

Figure 4. The process of encoding methods: see the context for details.

3.3 The learning vector quantization networks

A LVQ network [22] has a first competitive layer and a second linear layer. The linear layer transforms the competitive layer's classes into target classifications defined by the user. We refer to the classes learned by the competitive layer as subclasses and the classes of the linear layer as target classes. Both the competitive and linear layers have one neuron per (sub or target) class. Thus, the competitive layer can learn up to *S1* subclasses. These, in turn, are combined by the linear layer to form *S2* target classes. (*S1* is always larger than *S2*.) For example, suppose neurons 1, 2, and 3 in the competitive layer all learn subclasses of the input space that belongs to the linear layer target class No. 2. Then competitive neurons 1, 2, and 3, will have linking weight$_{2,1}$ weights of 1.0 to neuron n2 in the linear layer, and weights of 0 to all other linear neurons. Thus, the linear neuron produces a 1 if any of the three competitive neurons (1, 2, and 3) wins the competition and outputs a 1. This is how the subclasses of the competitive layer are combined into target classes in the linear layer.

The steps of LVQ1 algorithm are

Step 1. Initialize all weight vectors $w_j(0)$, learning rate parameter $\mu(0)$, and set $k=0$.

Step 2. Check the stopping condition. If false, continue; else if true, quit.

Step 3. For each training vector x_i perform steps 4 and 5:

Step 4. Determine the weight vector index $(j=q)$ such that

$$\min_{\forall j}\left\|x_i - w_j(k)\right\|_2^2 .$$

Step 5. Update the appropriate weight vector $w_q(k)$ as follows:

If $C_{w_q} = C_{x_i}$ then

$$w_q(k+1) = w_q(k) + \mu(k)\left[x_i - w_q(k)\right]$$

If $C_{w_q} \neq C_{x_i}$ then

$$w_q(k+1) = w_q(k) - \mu(k)\left[x_i - w_q(k)\right]$$

Step 6. Set $k \leftarrow k+1$, and reduce the learning rate parameter, then go to step 2.

4. Simulation results and discussions

The training data set [19] for this experiment consisted of 378 positive promoter patterns and 4500 negative randomly produced promoter patterns. The testing data set consisted of 50 positive and 500 negative promoter patterns. All the random negative patterns were excluded from the positive ones. The putative spacers for both the positive and negative patterns were caught though the EM algorithm. We extracted 35 bps from the training patterns, i.e., 17 bps for the -35 box, 11 bps for the -10 box, and 7 base pairs for the TSS. Finally, patterns to be learned by the neural networks were produced by the encoding methods.

As mentioned in a previous section, the length *k* of the extracted DNA sequence is 35 bps. Between the two choices (5 and 7), we chose *r*=5. Table 2 shows the experimental results of the purine-pyrimidine coded data set for the four different training methods. The four different training methods consisted of the LVQ network, the standard backpropagation (BP), the conjugate gradient (CG), and the Levenberg-Marquardt (LM) learning algorithm. According to Table 2, we found that the LVQ network learning has the best precision and specificity.

Table 2. Simulation results for the 9-dimension data set.

Methods	BP	LM	CG	LVQ
Precision (%)	90.41	88.91	88.55	90.78

We also applied the traditional encoding method to the same problem. The DNA alphabet {**A**, **C**, **G**, **T**} was encoded as a 4-bit pattern of {**1000**, **0100**, **0010**, **0001**}. Assume the (extracted) DNA patterns are *k* bps. The encoded training data via conventional encoding method would have 4*k* dimensions each. Since the actual length of *E. coli* promoter is still unknown to date, the value of *k* should always be set to a number that is larger than 33, the maximum length appearing in a sequenced annotation data file. Applying the traditional encoding method, we obtained 150-dimension data set that was used. The number of hidden nodes for these neural networks used was set to 20. As shown in Table 3, we found that the precision of the 150-dimension encoding method is similar to the 9-dimension encoding method.

Table 3. Simulation results for the 150-dimension data set

Methods	BP	LM	CG	LVQ
Precision (%)	90.55	90.36	87.27	90.91

5. Conclusions and future works

When the transcription starting sites are given or known, our adopted EM algorithm and encoding method can precisely recognize and predict the RNA polymerase binding sites, that is, the minus 35 (35 nucleotides upstream of the transcriptional start site) box and minus 10 (10 nucleotides upstream of the transcriptional start site) box. Then the promoter region and putative sequences can be predicted.

Through the feature extraction EM algorithm, we can precisely locate the minus 35 and minus 10 boxes for the *E. coli* promoter prediction. Then we propose a brand new encoding method. Based on two classes of DNA sequences, purine and pyrimidine, we efficiently reduce the input dimensions for the training and learning tasks in large scale. The benefit of this purine-pyrimidine coding approach is that demand for memory space and computational is decreased. The simulation results also prove that our new approach can achieve results nearly equal to that of the traditional orthogonal coding method for lower dimension coded data.

Reviewing the past literatures, we found that almost all related studies take no more than 500 promoter patterns to train their prediction systems. We chose the three new compilations of *E. coli* K12 promoter prediction researches as our positive training data sets. These three training sets were 362, 441, and 421 positive patterns with 65, 80, and over 300 bps, respectively. Most of these patterns do not indicate where the promoter regions are. We believe that our spacers locating algorithm is very suitable for these kinds of data.

In the future, based on the ongoing sequencing and annotating of *E. coli* genome sequence, there are another four structural parts of the DNA sequence that could be useful in the promoter prediction problem, i.e., RBS (ribosome binding sites), starting and ending of ORF (open reading frames), and transcription terminator detection (stem-loop). After finishing all of these sub-problems, we believe that the goal of perfect *E. coli* promoter prediction can be achieved. We would then like to expand our scheme to all other microorganisms.

Acknowledgement

This research is supported by the Nan-Kai College under grant NK91-29-C1201-10.

References

[1] Genome Online Database, http://www.genomesonline.org/

[2] Blattner, F. R., Plunkett, G., Bloch, C. A., Perna, N. T., Burland, V., Riley, M., Col-

lado-Vides, J., Glasner, J. D., Rode, C. K., Mayhew, G. F., Gregor, J., Davis, N. W., Kirkpatrick, H. A., Goeden, M. A., Rose, D. J., Mau, B., and Shao, Y. 1997. The complete genome sequence of Escherichia coli K-12. *Science,* 277, 5331: 1453-1474.

[3] Abello, J., Pardalos, P. M., and Resende, M. G. C. 2001. *"Handbook of Massive Data Sets"*. Dordrecht: Kluwer Academic: 1141-1168.

[4] Lin, C. T. and Lee, C. S. 1999. "Neural Fuzzy Systems – *a Neuro-Fuzzy Synergism to Intelligent Systems"*. Singapore: Prentice-Hall.

[5] Pederson, A. G. and Engelbrect, J. 1995. Investigations of Escherichia coli promote sequences with artificial neural networks: new signals discovered upstream of the transcriptional startpoint. *Proceedings of 3rd International Conference on Intelligent Systems for Molecular Biology:* 292-299.

[6] Handley, S. 1995. Predicting whether or not a nucleic acid sequence is an E. coli promoter region using genetic programming, *Proceedings of 1st International Symposium on Intelligence in Neural and Biological Systems*.

[7] Hirsh, H. and Noordewier, M. 1994. Using background knowledge to improve inductive learning of DNA sequences. *Proceedings of IEEE Conference on Artificial Intelligence for Applications*.

[8] Wu, C. H. 1997. Artificial neural networks for molecular sequence analysis. *Computers and Chemistry*, 21, 4: 237-256.

[9] Stormo, G. D., Schneider, T. D., and Gold, L. 1982. Use of the perceptron algorithm to distinguish translational initiation sites in E. coli. *Nucleic Acid Research*, 10: 2997-3011.

[10] Dayhoff, J. 1990. *"Neural Network Architectures: an Introduction"*. New York: Van Nostrand Reinhold.

[11] Peng, C. C. and Lin, C. J. 2002. *E. coli* Promoter prediction using neural fuzzy networks. *Proceeding of 10th National Conference on Fuzzy Theory and Its Applications*, Hsinchu, Taiwan.

[12] Chamberlin, M. J. 1974. The selectivity of transcription. *Annual Review of Biochemistry*, 43, 0: 721-775.

[13] Hawley, D. K. and McClure, W. R. 1982. Mechanism of activation of transcription initiation from the lambda PRM promoter. *Journal of Molecular Biology*, 157, 3: 493-525.

[14] Hawley, D. K. and McClure, W. R. 1983. The effect of a lambda repressor mutation on the activation of transcription initiation from the lambda PRM promoter. *Cell*, 32, 2: 327-333.

[15] Schneider, T. D. and Stephens, R. M. 1990. Sequence logos: A new way to display consensus sequences. *Nucleic Acids Research*, 18, 20: 6097-6100.

[16] Newlands, J. T., Josaitis, C. A., Ross, W., and Gourse, R. L. 1992. Both fis-dependent and factor-independent upstream activation of the rrnB P1 promoter are face of the helix dependent. *Nucleic Acids Research*, 20, 4: 719-726.

[17] Ross, W., Gosink, K. K., Salomon, J., Igarashi, K., Zou, C., Ishihama, A., Severinov, K., and Gourse, R. L. 1993. A third recognition element in bacterial promoters DNA binding by the alpha subunit of RNA polymerase. *Science*, 262, 5138: 1407-1413.

[18] Busby, S. and Ebright, R. H. 1994. Promoter structure, promoter recognition and transcription activation in prokaryotes. *Cell*, 79, 5: 743-746.

[19] Ozoline, O. N., Deev, A. A., and Arkhipova, M. V. 1997. Non-canonical sequence elements in the promoter structure. Cluster analysis of promoters recognized by E. coli RNA polymerase. *Nucleic Acids Research*, 25, 33: 4703-4709.

[20] Wu, C. F. J. 1983. On the convergence properties of the EM algorithm. *The An-*

nals of Statistics, 11, 1: 95–103.

[21] Ash, R. 1965. "*Information Theory*". New York: Interscience.

[22] Kohonen, T. 1987. "*Self-Organization and Associative Memory*". 2nd Edition, Berlin: Springer-Verlag.