Markov Encoding for Detecting Signals in Genomic Sequences

Jagath C. Rajapakse and Loi Sy Ho

Abstract—We present a technique to encode the inputs to neural networks for the detection of signals in genomic sequences. The encoding is based on lower-order Markov models which incorporate known biological characteristics in genomic sequences. The neural networks then learn intrinsic higher-order dependencies of nucleotides at the signal sites. We demonstrate the efficacy of the Markov encoding method in the detection of three genomic signals, namely, splice sites, transcription start sites, and translation initiation sites.

Index Terms—Genomic sequences, gene structure prediction, Markov chain, neural networks, splice sites, transcription start site, translation initiation site.

1 Introduction

Living organisms carry genetic information in the form of DNA molecules. Recent advances in DNA sequencing technology have led to an explosion of genomic data. Information in cells passes from DNA to mRNA to proteins through processes called *transcription* and *translation*. Each DNA molecule contains genes which decide the structural components of cells, tissues, and enzymes for biochemical reactions essential for its survival and functioning. In the process of transcription, genes in the DNA sequences are converted into corresponding mRNA sequences. In the process of translation, the nucleotides in the coding regions are translated to synthesize proteins. In eukaryotic genomes, a gene is structured by a variety of biological features, such as promoter, start codon, introns, exons, splice sites, and stop codon.

Signals in genomic sequences refer to specific sites or small sequence segments that are directly related to transcription and translation processes or to their regulation. This paper deals with computational techniques that identify signals in genomic sequences. Knowledge of the presence of signals in genomic sequences gives insight into transcription and translation processes and the location and annotation of genes, that are vital to the investigations of novel and effective drugs having minimal side effects. We address two important problems in signal detection, how to automatically identify the *transcription start sites* (TSS) and recognize the *translation initiation sites* (TIS) in eukaryotic DNA sequences, and another important problem in gene annotation: the determination of intron and exon boundaries or *splice sites* (SS). Bioinformatics approaches for automatic annotation of genomic sequences have recently gained increased attention because of the capital-intensive and time-consuming nature of pure experimental approaches.

• *The authors are with the BioInformatics Research Center, School of Computer Engineering, Nanyang Technological University, Singapore 639798. E-mail: asjagath@ntu.edu.sg, hsl@pmail.ntu.edu.sg.*

Manuscript received 31 Aug. 2004; revised 29 Nov. 2004; accepted 13 Dec. 2004; published online 2 June 2005.
For information on obtaining reprints of this article, please send e-mail to: tcbb@computer.org, and reference IEEECS Log Number TCBBSI-0103-0804.

1.1 Motivation

Though many attempts have been made to localize signals with appropriate models and algorithms, using genetic contexts, every technique has its own limitations and drawbacks [33]. Accurate detection of regulatory signals, such as SS, TSS, or TIS, needs to take into account the underlying relationships among nucleotides or features surrounding them. Though the potential of higher-order Markov models has been touted to represent complex interactions among genomic sequences, their implementation has become practically prohibitive because of the need for the estimation of large numbers of parameters [18]. The neural network approaches, on the other hand, are capable of discriminating intrinsic patterns in the vicinity of signals by finding appropriate non-linear mapping. The networks that receive low-level inputs, e.g., a string of nucleotides, do not employ explicit sequence features of biological significance. Combining the probabilistic and neural network approaches in a sensible way would lead to more efficient approaches in modeling and detecting signals.

This paper introduces the Markov/neural hybrid approach to signal detection by introducing a novel encoding scheme for inputs to neural networks, using lower-order Markov chains. The lower-order Markov models incorporate biological knowledge differentiating the compositional properties of the regions surrounding the signals; the neural networks combine the outputs from Markov chains, which we refer to as *Markov encoding* of inputs, to derive long-range and complex interactions among nucleotides, that improve the detection of signals. We will demonstrate the ability and efficacy of our approach in the detection of SS, TSS, and TIS. The significant improvements of the accuracies achieved for signal detection with Markov encoding would suggest that the lower-order Markov chains correctly represent the input features relevant to biological phenomena and the proposed Markov/neural hybrid systems have the potential for representing complex signals in genomic sequences.

2 Previous Approaches for Signal Detection

2.1 Splice Sites (SS)

Considering 5′/3′ direction, a *donor site* is the SS at the 5′ end and an *acceptor site* is the SS at the 3′ end of an intron. The splicing machinery needs to recognize and remove introns to make the correct message for protein production, yielding the importance of locating the SS in genomic sequences. SS have some distinct characteristics; one such characteristic is the "AG-GT" rule that an acceptor site has a conserved AG di-nucleotide and a donor site has a conserved GT di-nucleotide. An exception to this rule is found in a small handful of cases which have AC and AT di-nucleotides at either end of the sites instead [4]. Unfortunately, the detection of SS is often complicated because of common occurrences of consensus di-nucleotides at sites other than the SS.

Various computational techniques and algorithms have recently been developed for SS detection: neural network approaches [4], [11], [14], [29], [34], probabilistic models [6], [26], [38], and techniques based on discriminant analysis [39]. These methods primarily seek consensus motifs or features surrounding the SS by deriving a priori models from training samples [16]. Neural networks attempt to recognize the complex features of the neighborhood surrounding the consensus di-nucleotides by learning an intrinsic nonlinear transformation [4], [11], [14], [29], [34]. Probabilistic models, in a different way, estimate position-specific probabilities of nucleotides at the SS by computing likelihoods of the candidates of signal sequences [6], [26], [38]. The discriminant analysis uses several statistical measures to evaluate the presence of specific nucleotides, allowing recognition of the SS without explicitly determining their probability distributions [39]. Additionally, postprediction rule-based filtering techniques have often been performed empirically to enhance the performance of the prediction [14], [16]. Though a substantial number of techniques have been reported in recent literature, the accuracy of SS detection has not yet been satisfactory enough [10], [23], [35].

2.2 Transcription Start Sites (TSS)

A *promoter* is a region centered around a TSS which is biologically the most important signal controlling and regulating the initiation of the transcription of the gene immediately downstream [40]. A gene has at least one promoter [9]. Promoters have very complex sequence structures, reflecting the complicated protein-DNA interactions during which various transcription factors (TF) are bound [1]. They can be dispersed or overlapped, largely populating in about 1 Kbp region upstream and surrounding the TSS. Their functions can be either positive or negative and are often context-dependent. Eukaryotes have three different RNA-polymerase promoters that are responsible for transcribing different subsets of genes: RNA-polymerase I promoters transcribe genes encoding ribosomal RNA, RNA-polymerase II promoters transcribe genes encoding mRNA and certain small nuclear RNAs, and RNA polymerase III promoters transcribe genes encoding tRNAs and other small RNAs [25]. The present work focuses on detecting TSSs in RNA polymerase II promoters whose regulation is the most complex of the three types.

The typical structure of a polymerase II promoter is approximately located in the region $[-50, +50]$ with respect to the TSS and contains multiple binding sites that occur in specific contexts. The TF binding sites are typically 5-15 bp long [9], [25]. The nucleotide specificity at different positions within the sites varies. A TATA box, a binding site usually found at -25 bp upstream of the TSS in metazoans, is the most conserved sequence motif in the core region [7]. Around the TSS, there is a loosely conserved initiator region, abbreviated by Inr, which overlaps with TSS. The Inr is a much weaker signal compared to the TATA box, but also an important determinant of promoter strength. In the absence of TATA box, the Inr could determine the location of the TSS [9]. Also, there exist many regions giving exact matches to the TATA box and the Inr which do not represent the true promoter regions.

In spite of a number of computational techniques and algorithms developed for promoter recognition, only the applications of probabilistic models [7], [21], [22] and neural networks [30] have reported a fair degree of success [9]. As the promoter activity depends closely on how different TFs bind to the promoter region, in principle, the binding sites can be determined by looking at the contextual dependencies of the nucleotides. Neural networks attempt to recognize the complex features of the neighborhood surrounding the TSS by learning the arbitrary nonlinear transformations [30]. Probabilistic models, in a different way, focus on the homology search in the vicinity of the promoter by estimating position-specific probabilities of elements [1]. An evolutionary algorithm evolving straightforward motifs in the promoter region has recently been applied to locate potentially important patterns by inspection [7]. One can alternatively combine modules recognizing individual binding sites by using some overall description on how these sites are spatially arranged [9].

2.3 Translation Initiation Sites (TIS)

The beginning of the coding region of a gene is marked by a TIS, most of which contain a conserved triplet of nucleotides ATG, or the *start codon*, while a few exceptions are reported in eukaryotes [12], [24], [37]. At the first stage of the translation process, a small subunit of ribosome binds at the capped 5′-end of the mRNA and subsequently scans through the sequence until the first ATG in appropriate context is encountered [17]. The first occurrence of triplet ATG in the sequence may not always be chosen for translation as in approximately 40 percent of cases, an ATG further downstream is reported to be selected [24]. The TIS detection becomes more complex when using unannotated genomes that are not fully understood or analyzing expressed sequence tags which usually contain errors [18].

Analyses to uncover underlying features in the vicinity of TIS that are important for discovering true sites have been explored by using a number of computational techniques and algorithms. From a weight matrix estimated on an extended collection of data, Kozak [17] derived the consensus motif $gcc\underline{a}ccATG\underline{g}$, in which the position $+4$ (nucleotide G) and position -3 (nucleotide A) are highly conserved and exert the strongest effect, assuming that +1 is the position of nucleotide A of the conserved triplet ATG. This result was later enhanced by Pedersen and Nielsen [24] when they trained a feedforward multilayer neural network with input windows of 203 nucleotides centered at the ATG, except for one disregarded position, and revealed that position -3 is crucial for TIS recognition. They also

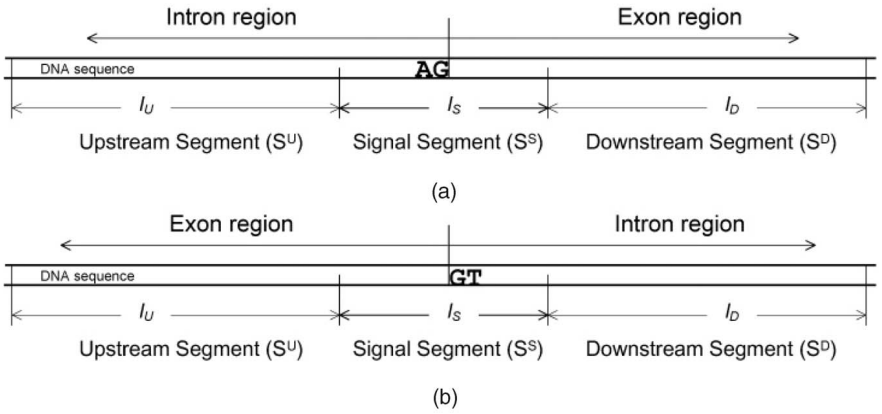

Fig. 1. The representation of a splice site by a signal segment and upstream segment and downstream segments of DNA: (a) acceptor site and (b) donor site.

discovered that ATGs that are in-frame to the TIS are more likely to be predicted incorrectly as TIS regardless of whether they are upstream or downstream of the start codon. Hatzigeorgiou developed an integrated method by combining a consensus neural network sensitive to the conserved motif and a coding neural network sensitive to the coding-noncoding potential around the start codon, with a ribosome scanning model [12]. While the consensus neural network assesses a window of nucleotides from position -7 to +5 relative to the triplet ATG, the coding neural network working on a window of 54 nucleotides applies to the codon usage statistic. Salamov et al. [31] showed that the most important component for the correct identification is the positional triplet weight matrix around conserved triplet ATG and the hexanucleotide difference before and after the ATG in a window of 50 bases long. The results of the locality-improved kernel in Zien et al. [41] over that of standard polynomial kernels suggested that the local correlations are more important than long-range correlations in identifying TIS. Wong et al. [18], [37] proposed a data mining tool comprised of generating and selecting explicit features and integrating the selected features for decision making; their work found nine relevant features.

3 Models of Signals

3.1 Markov Models of DNA Sequences

Segments of genomic sequences are often modeled by Markov models whose observed state variables are elements drawn from the alphabet Ω_{DNA} of four bases: A, T, G, and C. The Markov chain is defined by a number of states, equal to the number of nucleotides in the sequence, where each state variable of the model corresponds to a nucleotide. Consider a sequence $(s_1, s_2, \ldots s_l)$ of length l, modeled by a Markov chain, where the nucleotide $s_i \in \Omega_{DNA}$ is a realization of the ith state variable of the Markov chain. Except from state i to state $i+1$, there is no transition from state i to the other states. The model serially travels from one state to the next while emitting letters from the alphabet Ω_{DNA} in which each state is characterized by a position-specific probability parameter. In previous work [1], [6], [32], the Markov chain model is referred to as weight array matrix model.

If the Markov chain, say M, has an order k, the likelihood of the sequence is given by

$$P(s_1, s_2, \ldots s_l|M) = \prod_{i=1}^{l} P_i(s_i), \tag{1}$$

where the Markovian probability $P_i(s_i) = P(s_i|s_{i-1}, s_{i-2}, \ldots s_{i-k})$ denotes how conditionally the appearance of the nucleotide at location i depends on its k predecessors. In case $i \leq k$, lower-order dependencies should appropriately be used. Such a model is characterized by a set of parameters, $\{P_i(s_i|s_{i-1}, \ldots, s_{i-k}) : s_i, s_{i-1}, \ldots s_{i-k} \in \Omega_{DNA}, i = 1, 2, \ldots, l\}$. That is, we need to maintain 4^{k+1} parameters at each state to represent the segment.

3.2 Model of SS

The SS model is considered as a concatenation of three consecutive DNA segments: signal segment (S^S), upstream segment (S^U), and downstream segment (S^D), as illustrated in Fig. 1. The signal segment representing the consensus pattern responsible for splicing mechanism consists of nucleotides immediately neighboring the SS. The upstream and downstream segments adjoining the signal segment on both sides capture the features and contrast of the coding and noncoding sequences. If the length of the signal segment of the sequence at SS is l_S and the lengths of the upstream and downstream segments are l_U and l_D, respectively, the model of the splice site is represented by a sequence of length $l_U + l_S + l_D$. For the selection of the values of the lengths of these segments, the reader is referred to [4], [6], [26].

The three segments in the model of SS are represented by three Markov chains and a feedforward multilayer neural network receiving the Markovian probabilities as inputs. The Markovian probabilities are concatenated as illustrated in Fig. 2 and fed to the network with n input nodes: $n = l_U + l_S + l_D$. If the input to the jth input node of the network is x_j,

$$x_j = \begin{cases} P_i^U(s_i), & \text{if } j \leq l_U \text{ and } i = j; \\ P_i^S(s_i), & \text{if } l_U \leq j < l_U + l_S \\ & \text{and } i = j - l_U; \\ P_i^D(s_i), & \text{if } l_U + l_S \leq j < l_U + l_S + l_D \\ & \text{and } i = j - l_U - l_S. \end{cases} \tag{2}$$

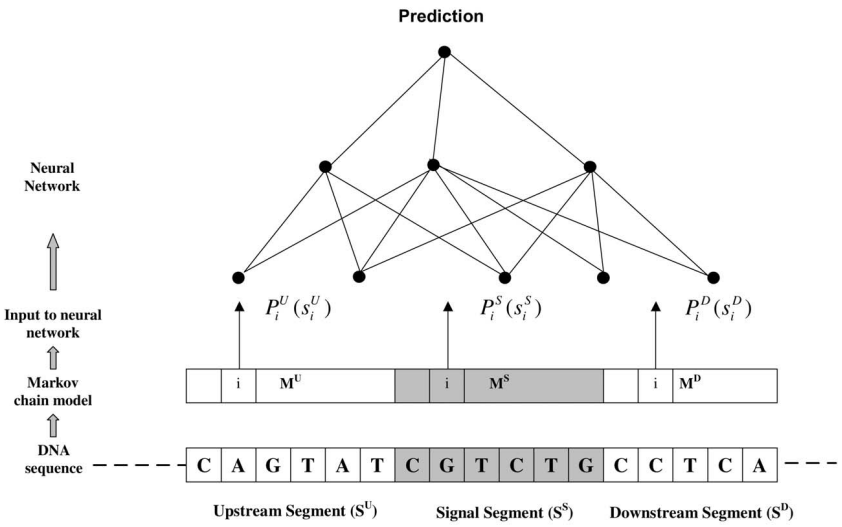

Fig. 2. Splice site prediction: The outputs of lower-order Markov models representing the three segments are processed by a multilayer neural network.

A first-order Markov chain is used to represent the consensus segment of the signal and second-order Markov chains are used to represent the upstream and downstream segments. The neural network combines the lower-order interactions of nucleotides, given by the Markov models, nonlinearly, to capture any relevant higher-order contextual information for SS prediction. Accordingly, the present hybrid model allows the incorporation of biological information as well as intrinsic distant interactions of nucleotides.

3.3 Model of TSS

The model of TSS is represented by a promoter of length 100 bp $[-50, 50]$, presuming that the candidate TSS position starts at $+1$. The promoter consists of two sequence segments, as illustrated in Fig. 3: the TATA-box segment, a subsequence of 30 bp from -40 to -10, and the Inr segment, a subsequence of 25 bp from -14 to 11. The segment sizes are selected so that the consensus motifs for both binding sites are included. For information on the rationale for the selection of the configuration of these segments, the reader is referred to [1], [22], [30]. There are three independent Markov chains modeling the TATA-box segment, the Inr segment, and the concatenation of the TATA-box and Inr segments. The concatenated segment allows long-range interactions between both sides of the TSS. A second-order is chosen for the TATA-box model while the first-order is chosen for the Inr model because of the weak consensus of the Inr segment [9].

The outputs of the Markov chains are processed by three parallel neural networks working in cascade, as illustrated in Fig. 4. The orthogonal coding of DNA sequences has often been used at the inputs of neural networks, which encodes each nucleotide s_i by four binary digits, say $\mathbf{b}_{s_i}$, where only one digit is set to 1 and the remaining digits are set to 0, for instance: $\mathbf{b}_A = (1,0,0,0)$, $\mathbf{b}_C = (0,1,0,0)$, $\mathbf{b}_G = (0,0,1,0)$, and $\mathbf{b}_T = (0,0,0,1)$. Given a sequence $(s_1, s_2, \ldots, s_l)$ of length l, where $s_i \in \Omega_{DNA}$, the orthogonal coding results in a vector $(\mathbf{b}_{s_1}, \mathbf{b}_{s_2}, \ldots, \mathbf{b}_{s_l})$ representing $4l$ number of inputs to the neural network. The Markov encoding combines the Markovian probability parameters with the orthogonal coding: the input sequence results in a vector $(\mathbf{c}_{s_1}, \mathbf{c}_{s_1}, \ldots, \mathbf{c}_{s_l})$ of inputs to the neural network where $\mathbf{c}_{s_i} = P_i(s_i)\mathbf{b}_{s_i}$, for $1 \le i \le l$. The Markov encoding thereby incorporates the homology of potential promoter regions, given by the Markov models, into the neural network-based approaches.

3.4 Model of TIS

The TIS model consists of three parts: conserved ATG triplet, an upstream segment, and a downstream segment, as illustrated in Fig. 5. The upstream segment consists of 50 nucleotides, from -50 to -1, and the downstream segment consists of 50 nucleotides, from $+4$ to $+53$, assuming that $+1$ is the position of the first nucleotide A of the candidate ATG. For the selection of the width of these segments, the reader is referred to [18], [24], [31], [41]. As the TIS without the conserved triplet ATG is reported to be rare in eukaryotes [37], such cases are not addressed here.

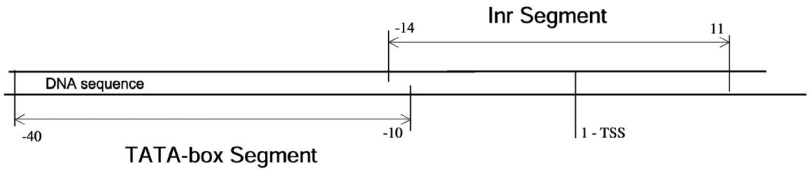

Fig. 3. The representation of the transcription start site (TSS) by a promoter consisting of a TATA-box and an Inr segment.

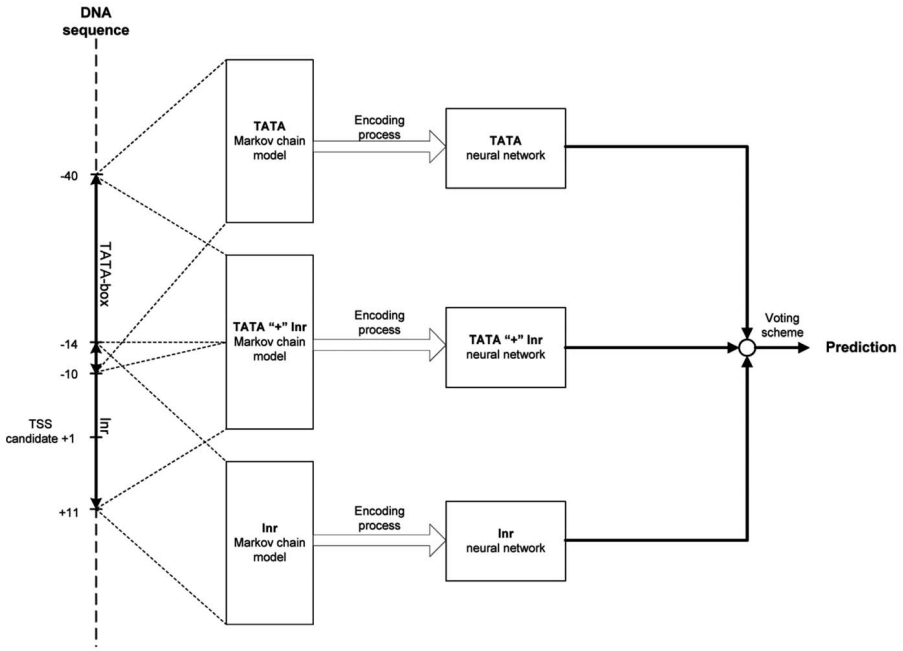

Fig. 4. Transcription start site (TSS) prediction: The outputs of three Markov chain models, representing the TATA-box segment, the concatenation of TATA-box and Inr segment, and Inr segment, are processed by three neural networks whose outputs are combined with a voting scheme.

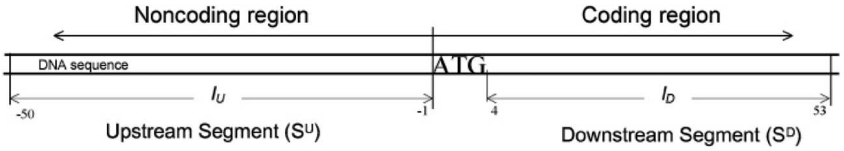

Fig. 5. The representation of a translation initiation site (TIS) by a conserved triplet, ATG, and upstream and downstream segments.

Two independent Markov chains model the upstream and downstream segments. Due to weak dependencies of the nucleotides in the noncoding region [32], the first-order is chosen for the upstream model while, in an attempt to capture codon distributions, the second-order is chosen for the downstream model. The output Markov probabilities are combined with orthogonal encoding in a manner similar to that used for input coding in the TSS detection. In a real eukaryotic genome, the possibility of a true TIS signal without a conserved triplet ATG is much lower than the possibility of a true SS without a conserved di-nucleotide. Therefore, we decided to avoid ATG start codon in the concatenation of the upstream and downstream segments.

The downstream segment is further represented by a protein encoding technique based on a coding differential analysis [6], [12], [32]. In order to account for a coding measure in the downstream region, the protein encoding model transforms the DNA sequence to a vector of 38 elements by applying Percent Accepted Mutation (PAM) matrices [8]. The nucleotide sequences are first converted into amino acid sequences and then the amino acids are grouped into six exchange groups, representing conservative replacements through evolution [36]; the 36 elements of the vector give normalized frequencies of the corresponding di-exchange symbols and the last two elements track the presence of stop codon in the downstream segment. Previous investigations have shown that the coding differential method is effective in distinguishing coding from noncoding regions [6], [12], [32].

The outputs from Markov chains of upstream and downstream segments and the protein coding model for downstream segments are processed by three feedforward neural networks, as illustrated in Fig. 6. The outputs of three neural networks are combined using a majority voting scheme for the prediction. The approach searches the TIS in the DNA sequence until at least two of three neural networks predict truly. Once the first suitable ATG is detected as the correct TIS, the present method stops scanning the DNA sequence for another TIS as in the ribosome scanning model [12].

4 Higher-Order Markov Model of Signals

4.1 Neural Networks

The neural networks used in the models were multilayer perceptron (MLP). If an MLP network has n input nodes, one hidden-layer of m neurons, and one output neuron, the output of the network is given by

$$y = f\left(\sum_{k=1}^{m} w_k f_k\left(\sum_{j=1}^{n} w_{kj} x_j\right)\right), \tag{3}$$

where $f_k,\ k = 1, 2, \ldots, m$, and f denote the activation functions of the hidden-layer neurons and the output neuron,

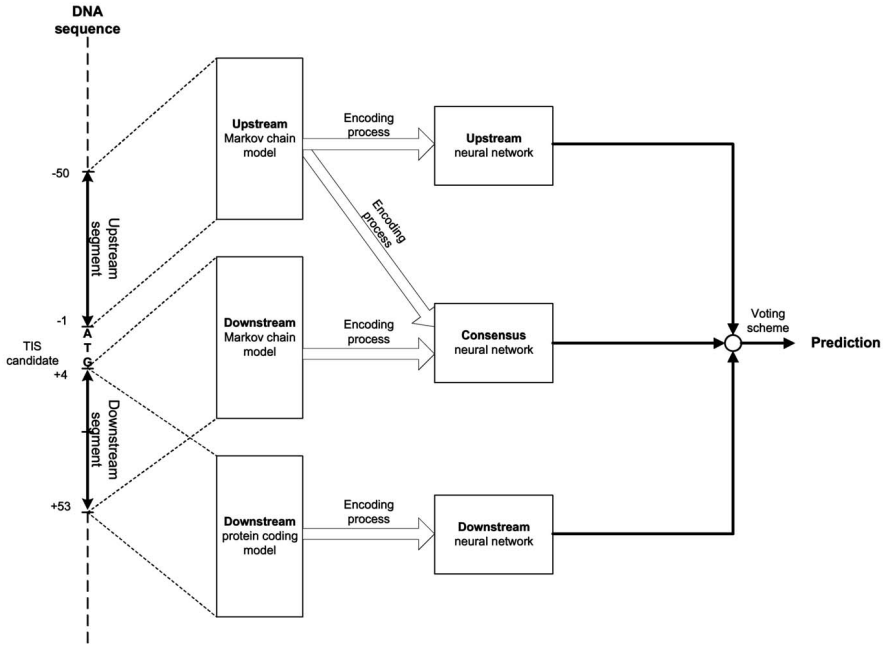

Fig. 6. Translation initiation site (TIS) prediction: The outputs of the upstream and downstream Markov chain models and the downstream protein encoding model are processed by three neural networks whose outputs are combined in a voting scheme.

respectively; $w_k,\ k = 1, 2, \ldots m$ and $w_{kj},\ k = 1, 2, \ldots m, j = 1, 2, \ldots n$ denote the weights connected to the output neuron and to the hidden-layer neurons, respectively. The output activation function was selected to be a unipolar sigmoidal: $f(u) = \alpha/(1 + e^{-\beta u})$ and the hidden-layer activation functions took the form of hyperbolic tangent sigmoidals: $f_k(u) = \alpha_k(e^{\beta_k u} - e^{-\beta_k u})/(e^{\beta_k u} + e^{-\beta_k u})$, for all k. The weights of the networks were learned with the standard backpropagation algorithm [13].

4.2 Higher-Order Markov Models of Signals

Lower-order Markov chains provide a means of incorporating the characteristics of coding and noncoding regions by linearly modeling interactions among neighboring nucleotides. The neural networks in our approach receive Markovian probabilities and combine them nonlinearly in order to learn more complex and distant interactions among nucleotides. In what follows, we show that, indeed, such lower-order Markov/neural hybrid models result in higher-order Markov models of signals.

According to Ohler et al. [21], given a subsequence $\mathbf{s}_1^i = (s_1, s_2, , s_i)$, conditional dependencies can be approximated by rational interpolation:

$$P(s_i|\mathbf{s}_1^{i-1}) \approx \frac{\sum_{k=0}^{i-1} a_k g_k(\mathbf{s}_{i-k}^{i-1}) \widehat{P}_i(s_i|\mathbf{s}_{i-k}^{i-1})}{\sum_{k=0}^{i-1} a_k g_k(\mathbf{s}_{i-k}^{i-1})}, \tag{4}$$

where $\widehat{P}_i(\cdot)$ denotes the empirical probability obtained from Maximum Likelihood (ML) estimates from training data sets, coefficient a_ks denote real value coefficients such that $\sum_{k=0}^{i-1} a_k = 1$, and g_ks denote functions that represent the relationships of different-order contextual interactions among variables $s_k,\ k = 1, 2 \ldots i$ [21].

By using the chain rule of probabilities, the likelihood of the sequence $\mathbf{s}_1^l$ is given by

$$P(s_1, s_2, \ldots s_l) = P(s_1) \prod_{i=2}^{i} P(s_i|\mathbf{s}_1^{i-1}). \tag{5}$$

By induction of (4), that is, by replacing conditional probabilities with the probabilities conditioned by a smaller number of elements, we obtain

$$P(s_1, s_2, \ldots s_l) \approx P(s_1) \prod_{i=2}^{l} \sum_{j=1}^{i-1} b_{ij} \widehat{P}_i(s_i|\mathbf{s}_{i-j}^{i-1}), \tag{6}$$

where $\{b_{ij} : i = 2, \ldots l, j = 1, \ldots i\}$ is a set of real numbers. That is, the nonlinear relationship among variables in the sequence is represented approximately by polynomials of sufficient order.

As shown by Pinkus [28] and Hornik et al. [15], the output of the neural network is capable of representing the input-output relationship with a sufficiently large higher-order polynomial. Hence, the output of a neural network receiving Markov probabilities $P_i(s_i),\ i = 1, 2, \ldots l$, can be written as

$$y = \sum_{m_1, \ldots m_l = 0; m_1 + \ldots m_l = l} c_{m_1, \ldots m_l} P_1(s_1)^{m_1} \ldots P_l(s_l)^{m_l}, \tag{7}$$

where $\{m_i; i = 1, 2, \ldots l\}$ is a set of nonnegative integers and $\{c_{m_1, \ldots, m_l}; m_1 + \ldots m_l = l\}$ is a set of real value coefficients. By observation of (6) and (7), it can be deduced that the neural network output, y, represents a higher-order Markov model that takes care of all the conditional interactions among all the nucleotides in the input sequence.

5 Experiments and Results

In this section, we demonstrate the performance of our approach, using experiments with benchmark data sets and comparisons with earlier approaches. Experiments were

evaluated based on the standard performance measures: sensitivity, specificity, precision, and accuracy.[1] The sensitivity gives the percentage of correct prediction of true sites and the specificity gives the percentage of correct prediction of false sites. Whenever possible, we obtain Receiver Operation Characteristics (ROC) plotting the sensitivity values against one minus specificity at various threshold values at the output of neural networks.

5.1 Training

The prediction models were trained in two phases: The Markov chains' parameters were estimated first and, thereafter, the neural networks' training was done. In order to evaluate Markov model parameters, the corresponding sequence segments were first aligned. The ML estimate of the ith state of the k-order Markov model, say $\widehat{P}_i(\cdot)$, is given by the ratios of the frequencies of all partial sequences of $k+1$ elements at i and k elements at $i-1$ positions:

$$\widehat{P}_i(s_i) = \frac{\#(\mathbf{s}_{i-k}^{i})}{\#(\mathbf{s}_{i-k}^{i-1})}, \tag{8}$$

where $\#(\cdot)$ represents the frequency of its argument in the training data set.

In SS detection, for finding the parameters of the model for true sites, the training sequences were aligned with respect to the consensus di-nucleotide. The TATA-box and Inr segments were extracted from the scanning window in the TSS detection and the Markov model parameters were estimated using the frequencies of nucleotides at each position. In TIS detection, at each ATG site, 50 bp upstream and 50 bp downstream (without ATG) were extracted, resulting in 50 bp long sequences which may have flanking N regions because of the lack of data in the upstream context and/or downstream context of the ATG site.

Once the Markov chain parameters were learned, the training sequences were again applied to the model and the Markovian probabilities were used as inputs to neural networks. Desired outputs were set to either 0.9 or 0.1 to represent the true or false site at the output, correspondingly. In order to avoid overfitting, the number of the hidden neurons was initially set such that the total number of free parameters, i.e., the weights and biases, should be η times the number of the training sequences, where η denotes the fraction of detection errors permitted on testing [13]. Starting from this configuration, the optimal number of hidden neurons was then determined empirically.

For all the neural networks, the parameters of the activation functions were $\alpha = 1.0$, $\beta = 1.716$, and $\alpha_k = 1.0$ and $\beta_k = 1.0$ for all k; we noticed that the performance did not depend much on slight variations of these parameter values. The standard online error backpropagation algorithm was used to train the neural networks [13] with momentum initially set to 0.01 and learning rate initially set to 0.1, for each hidden and output neuron, and updated iteratively as in [19], [27]. The weights of neural networks were initialized in [-1.0, 1.0] using the Nguyen-Widrow method [20].

1. Sensitivity $= \frac{TP}{TP+FN}$, specificity $= \frac{TN}{TN+FP}$, precision $= \frac{TP}{TP+FP}$, and accuracy $= \frac{TP+TN}{TP+TN+FP+FN}$, where TP, TN, FP, and FN denote the rates of true positives, true negatives, false positives, and false negatives, respectively [5], [37].

5.2 SS Detection

In this section, we compare the performance of our SS detection method on two data sets: NN269 available at [46] and GS1115 prepared by Pertea et al. [26]. In the case of acceptor sites, a 189 bp window around the consensus di-nucleotide (at the site) was used for all samples: 21 bp for the intron (ending with AG) and 8 bp for the following exon, i.e., $l_S = 29$, and two additional sequences of 80 bp, i.e., $l_U = l_D = 80$. In the case of donor sites, a 176 bp window around the consensus di-nucleotide (at the site) was used for all samples, which has 6 bp of the exon and 10 bp of the following intron (starting with GT), $l_S = 16$, and two additional sequences of 80 bp, i.e., $l_U = l_D = 80$. These asymmetric configurations were selected because the most conserved parts of the splicing signals extend more into the intron regions [11].

The data set NN269, available at [46], provided by Reese et al. [29] consists of 269 human genes. Confirmed 1,324 true acceptor sites and 1,324 true donor sites were extracted from the data set. Additionally, 5,552 false acceptor and 4,922 false donor sites with the consensus GT or AG di-nucleotides appearing in a neighborhood of plus and minus 40 nucleotides around a true splice site were collected. Given the above numbers, we used the empirical method, explained above to determine that five hidden neurons were optimal for both donor and acceptor networks.

Fig. 7 illustrates a comparison of the performance of the present SS prediction with NNSplice [45] and GeneSplicer [42], using the NN269 data set. We used the same experimental setup and data partitions that were used in the experimentation of NNSplice [45]: The entire data set was divided into a training set containing 1,116 true acceptor, 1,116 true donor, 4,672 false acceptor, and 4,140 false donor sites, and a test set containing 208 true acceptor, 208 true donor, 881 false acceptor, and 782 false donor sites. As seen, the present approach showed more power in the detection of splice sites than both the NNSplice and GeneSplicer.

Further, the SS prediction was tested using the data set GS1115 prepared by Pertea et al. [26], using five-fold cross-validation, to compare with the performance of GeneSplicer [42], a method based on a Markov model. We used the same data partions on GS1115 data set, used by GeneSplicer [42]: Together with confirmed true 5,733 acceptor sites and 5,733 donor sites extracted from 1,115 human genes, 650,099 false acceptor and 478,983 false donor sites, with confirmed GT or AG di-nucleotides present and that are not annotated as true sites, were collected. For this experiment, the neural networks showed optimal performance with 15 hidden neurons for the donor network and 30 hidden neurons for the acceptor network. The performances of the present approach with other previous techniques are given in Fig. 8. Only the GeneSplicer program allows retraining the data set and testing at various values of FN to obtain the complete ROC curve. The performances of HSPL [44], SpliceView [48], and GENIO [43] are only available at specific FN values. As the publicly available NNSplice program does

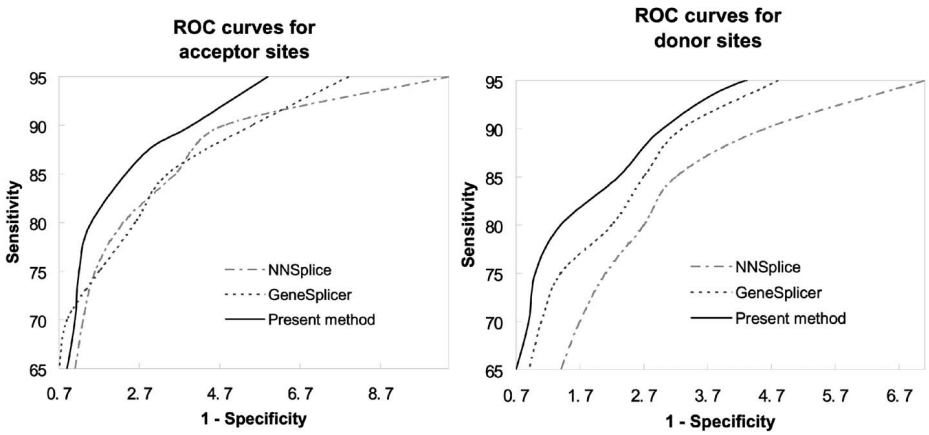

Fig. 7. Comparison of splice site (SS) detection by the present method, GeneSplicer, and NNSplice: ROC curves on the nonredundant 269 human gene data set (NN269) for acceptor sites and donor sites.

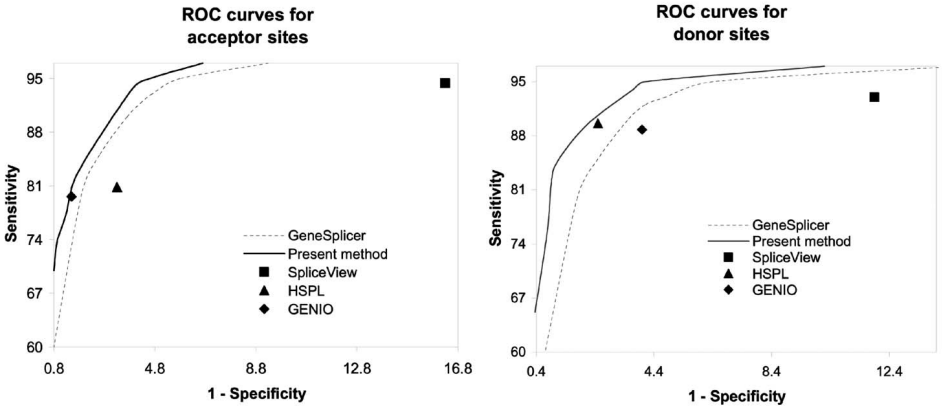

Fig. 8. Comparison of performance of splice site (SS) detection with the GeneSplicer: ROC curves on the nonredundant 1,115 human gene data set (GS1115) for acceptor sites and donor sites.

not allow retraining with new data sets, the comparison with NNSplice is not presented.

Although the training of Markov chain models was in real time, the training of the neural network was relatively slow when there was a massive imbalance of the number of true sites and false sites in the training data set. Therefore, we selected a set of equal true and false ones, randomly in every epoch, to train the neural network to improve the training speed. A refinement step was performed in the postprediction for further eliminating wrong predictions and enhancing weak predictions, taking into consideration 1) the alternative positions of acceptors and donors [14], 2) the compositional differences between exons and introns [6], and 3) the maximum and minimum lengths of introns and internal exons [26].

Fig. 9 displays the probability distribution of the nucleotides at specific positions in a representative upstream segment $[-86, -7]$ and downstream segment $[+11, +90]$ of a true donor site represented by the second-order Markov model and the die model (zero-order Markov model) [2], assuming that the conserved GT di-nucleotide is labeled $[+1, +2]$. The Markov models highlight the differences of positional probabilities of nucleotides better than the die model, taking into consideration the neighborhood dependencies such as codon distribution. Fig. 10 illustrates how the Markov model captures the contrast between upstream and downstream segments.

5.3 TSS Detection

In this study, the human promoter data set provided by Reese [47] was used. The set essentially consists of three parts: promoter sequences, CDS (coding) sequences, and noncoding (intron) sequences. Each part contains five sequence sets to be used for five-fold cross-validation. The promoters were extracted from the Eukaryotic Promoter Database (EPD) release 50 (575 sequences). Promoter entries with less than 40 bp upstream and/or 5 bp downstream were discarded, leaving 565 entries, out of which 250 bp upstream and 50 bp downstream were extracted, resulting in 300 bp long sequences which may have flanking N regions due to the lack of data in the beginning and/or end of the promoter region. The false set contains coding and noncoding sequences from the 1998 GENIE data set [47]. The exons were concatenated to form single CDS sequences. The sequences were cut consecutively into 300 bp long nonoverlapping sequences. The shorter and remaining sequences at the end were discarded. Finally, 565 true promoters, 890 CDS sequences, and 4,345 intron sequences were included in the data set. The TATA neural network showed optimal performance with seven hidden neurons,

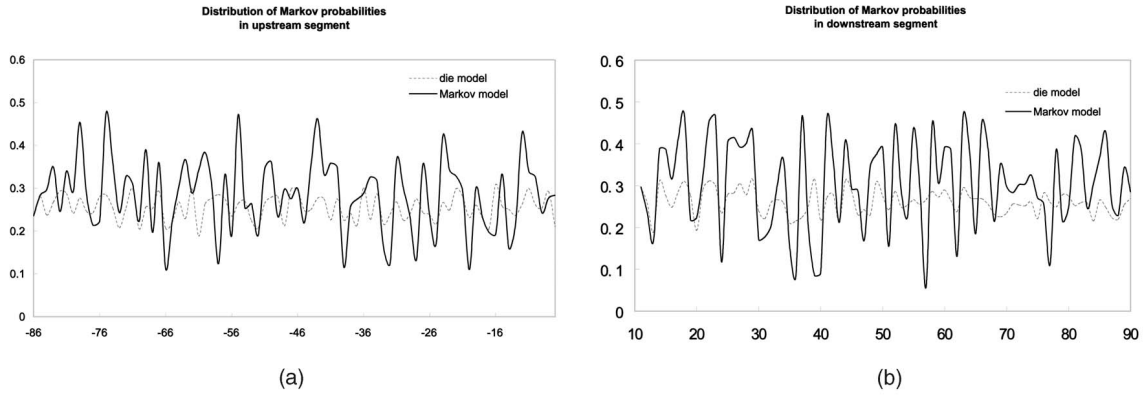

Fig. 9. Illustration of Markov encoding in a representative (a) upstream and (b) downstream segment of a true donor site by comparing to positional probabilities given by the die model.

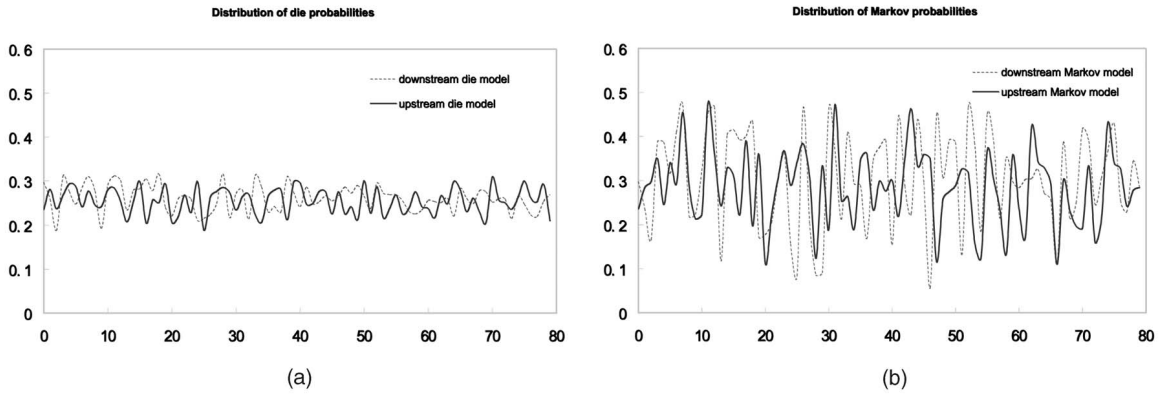

Fig. 10. Illustration of the distribution of parameters of (a) die models and (b) Markov models in the region surrounding a true donor site.

the Inr neural network with seven hidden neurons, and the concatenation of TATA and Inr network with 15 hidden neurons. Following NNPP 2.1 method [30], we used Time-Delay Neural Networks (TDNN), which is a variant of MLP whose hidden neurons activations are replicated over time [13], for TATA neural network.

Fig. 11 shows a comparison of performance of the detection of TSS between the present method using Markov encoding and the NNPP 2.1 method, which uses a TDNN receiving inputs from the concatenation of TATA and Inr segments, with orthogonal encoding. As seen, the ROC curves show the superiority of the present method over the NNPP 2.1 technique on the tested data set. We further used the standard correlation coefficient, r,[2] to evaluate our model. As seen in Table 1, the present method has a correlation coefficient rate of 0.69, on average, which is better than the rate 0.65 of NNPP 2.1.

5.4 TIS Detection

We used the data set provided by Pedersen and Nielsen for training, testing, and evaluation of the TIS detection [24]. The set consists of sequences extracted from GenBank release 95 and processed by removing introns and joining the remaining exon parts. From the resulting sequences, only those sequences containing at least 10 nucleotides upstream and 150 nucleotides downstream (relative to the A in the triplet ATG) were selected; all other sequences were discarded. The sequences were filtered to remove the redundancies because of the presence of genes belonging to gene families, homologous genes from different organisms, and sequences submitted to the database by more than one source. In total, 13,375 ATG sites were included into the data set, of which there are 3,312 true TISs and 10,063 false TISs. Of the false TISs, 2,077 are upstream of true TISs. The data set was split into three equal-sized subsets and used for three-fold cross-validation. For optimal performance, we empirically found five hidden neurons for the upstream neural network, seven hidden neurons for the downstream neural network, and 20 hidden neurons for the consensus neural network.

The performance comparisons between the present method and previous TIS recognition systems [12], [24], [18], [41] are illustrated in Table 2. The version of the data-mining program proposed in [18] used in this comparison was equipped with the ribosome scanning model. The result reported by Hatzigeorgiou [12] was not directly comparable since she used a different data set. As seen, the present method achieved an average 93.8 percent of sensitivity and 96.9 percent of specificity, compared to the previously reported best 88.5 percent of sensitivity and 96.3 percent of specificity by the data mining-based program [18].

2. The standard correlation coefficient [5]:

$$r = \frac{(TP \times TN) - (FN \times FP)}{\sqrt{(TP + FN) \times (TN + FP) \times (TP + FP) \times (TN + FN)}}. \quad (9)$$

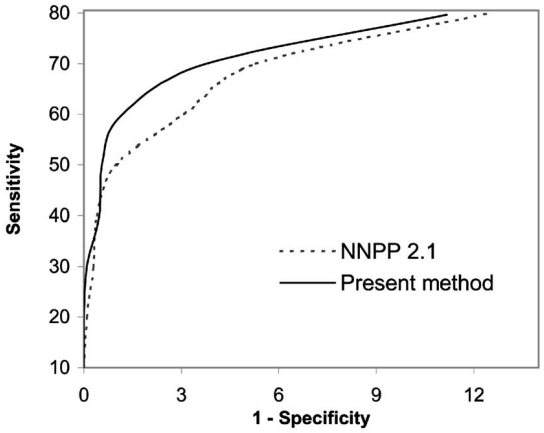

Fig. 11. Comparison of transcription start site (TSS) prediction between the present method and the NNPP 2.1 method.

6 Conclusion

We presented a Markov/neural hybrid approach to detect signals, such as SS, TSS, and TIS, in genomic sequences. The approach used Markov encoding: The inputs to the neural networks are Markovian probabilities that characterize the prior biological knowledge on coding and noncoding regions. The neural networks capture intrinsic features surrounding the signals by combining lower-order Markovian probabilities and finding an appropriate arbitrary mapping that represents the signal accurately. As shown, the present hybrid model effectively implements a complex higher-order Markov model. Although the higher-order Markov models were often touted earlier as accurate models to characterize signals [18], their direct implementation has been practically prohibitive because of the need for estimating the large number of parameters with the often limited amount of training data.

The lower-order Markov models incorporate useful biological knowledge such as the bias of codon distributions, the contrast between exons and introns, and the homology of potential signal sites. However, these models are incapable of representing more complex nonlinear interactions among nucleotides surrounding the signal sites, which may be useful for their identification. Neural networks, being large nonparametric nonlinear models, are capable of finding complex mapping through learning. In the present approach, the neural networks nonlinearly combine lower-order interactions, represented by Markov probabilities, to realize the higher-order distant interactions among nucleotides. Not only the lower-order Markov models, our method can be extended for other models to represent biological knowledge, such as the protein encoding model used in TIS prediction. Further, the Markov encoding scheme can be used as a preprocessing step for the inputs to the other detection techniques using Support Vector Machines, decision trees, etc., and, in principle, should improve their performances.

TABLE 1
Comparison of Correlation Coefficients, Averaged over Five-Fold Cross-Validated Sets between the Present Neural Network Method and NNPP 2.1 Method

%True Positives	NNPP 2.1 method	The Present Method
30	0.50	0.52
40	0.60	0.57
50	0.65	0.65
60	0.61	0.69
70	0.58	0.65
80	0.52	0.53
90	-	0.39

Notation "-" indicates that NNPP 2.1 was unable to achieve 90 percent true positives.

TABLE 2
Comparison of Average Results with Three-Fold Cross-Validation of the Present Method with Previous TIS Recognition Methods

	Sensitivity	Specificity	Precision	Accuracy
Wong *et al.* [18]	88.5%	96.3%	88.6%	94.4%
Pedersen *et al.* [24]	78.0%	87.0%	-	85.0%
Zien *et al.* [41]	69.9%	94.1%	-	88.1%
Hatzigeorgiou [12]	-	-	-	94.0%
Present method	93.8%	96.9%	90.8%	96.1%

Character "-" indicates that the values were not reported in the literature.

The tenet of any approach of signal detection is to first discover relevant biological features and then process them optimally for correct prediction with high sensitivity and specificity. The present method allows both incorporation of biological knowledge and learning of intrinsic complex features. Neural networks have been used earlier for prediction of signals with orthogonal coding [4], [11], [29], which did not allow the incorporation of prior knowledge. As demonstrated in the experiments, with the incorporation of explicit biological features, using the Markov models at the input, the accuracies of prediction of signals by the neural networks improved. Also, the present method outperformed the methods that use only Markov models, such as the GeneSplicer used in SS detection, as they are unable to capture more complex features resulting from nonlinear and distant interactions of nucleotides, and the pure neural network approaches, such as the NNPP 2.1 method in TIS detection, that use orthogonal encoding.

Our comparisons to the existing techniques were, however, limited due to the inaccessibility to the programs and the previously used data. On the different data sets with different characteristics, e.g., the proportion of true and false sites, the techniques gave different results. Nevertheless, the present method outperformed earlier approaches on the tested data sets, supporting that the proposed Markov/neural models can potentially model signals in genomic sequences, as illustrated by the detection of SS, TIS, and TSS. The parameters determined for neural network configurations and training might not be optimal, though our testing focused on investigating the effectiveness of Markov encoding method. As for the length of segments of signal models, we relied on previously reported values in the literature. Our future work involves investigation into the selection of the length of segments representing different signals under the present framework.

In conclusion, we proposed using lower-order Markov models for encoding the input sequences for the prediction of signals by neural networks and demonstrated the efficacy of the Markov/neural approach in the detection of three signals, namely, SS, TIS, and TSS. The present model is capable of implementing higher-order Markov models of the signal sites and provides an efficient and feasible method of learning model parameters, leading to better detection of the signals.

REFERENCES

[1] V.B. Bajic, S.H. Seah, A. Chong, S.P.T. Krishnan, J.L.Y. Koh, and V. Brusic, "Computer Model for Recognition of Functional Transcription Start Sites in RNA Polymerase II Promoters of Vertebrates," *J. Molecular Graphics and Modeling,* vol. 21, pp. 323-332, 2003.

[2] P. Baldi and S. Brunak, *Bioinformatics: The Machine Learning Approach,* first ed. MIT press, 1998.

[3] V. Brendel and J. Kleffe, "Prediction of Locally Optimal Splice Sites in Plant Pre-mRNA with Application to Gene Identification in *Arabidopsis Thaliana* Genomic DNA," *Nucleic Acids Research,* vol. 26, pp. 4748-4757, 1998.

[4] S. Brunak, J. Engelbrecht, and S. Knudsen, "Prediction of Human Mrna Donor and Acceptor Sites from the DNA Sequence," *J. Molecular Biology,* vol. 220, pp. 49-65, 1991.

[5] M. Burset and R. Guigo, "Evaluation of Gene Structure Prediction Programs," *Genomic,* vol. 34, pp. 353-367, 1996.

[6] C. Burge and S. Karlin, "Prediction of Complete Gene Structures in Human Genomic DNA," *J. Molecular Biology,* vol. 268, pp. 78-94, 1997.

[7] D. Corne, A. Meade, and R. Sibly, "Evolving Core Promoter Signal Motifs," *Proc. 2001 Congress on Evolutionary Computation,* pp. 1162-1169, 2001.

[8] M.O. Dayhoff, R.M. Schwartz, and B.C. Orcutt, "A Model of Evolutionary Change in Proteins," *Atlas of Protein Sequence and Structure,* vol. 15, no. 3, pp. 345-358, 1978.

[9] J.W. Fickett and A.G. Hatzigeorgious, "Eukaryotic Promoter Recognition," *Genome Research,* pp. 861-878, 1997.

[10] R. Guigo, P. Agarwal, J.F. Abril, M. Burset, and J.W. Fickett, "An Assessment of Gene Prediction Accuracy in Large DNA Sequences," *Genome Research,* vol. 10, pp. 1631-1642, 2000.

[11] A. Hatziggeorgious, N. Mache, and M. Reczko, "Functional Site Prediction on the DNA Sequence by Artificial Neural Networks," *Proc. IEEE Int'l Joint Symp. Intelligence and Systems,* pp. 12-17, 1996.

[12] A.G. Hatzigeorgiou, "Translation Inititation Start Prediction in Human cDNA with High Accuracy," *Bioinformatics,* vol. 18, pp. 343-350, 2002.

[13] S. Haykin, *Neural Networks: A Compreshensive Foundation,* second ed. Prentice-Hall Press, 1999.

[14] S.M. Hebsgaard, P.G. Korning, N. Tolstrup, J. Engelbrecht, P. Rouze, and S. Brunak, "Splice Site Prediction in Arabidopsis Thaliana Pre-mRNA by Combining Local and Global Sequence Information," *Nucleic Acids Research,* vol. 24, pp. 3439-3452, 1996.

[15] K. Hornik, M. Stinchcombe, and H. White, "Multilayer Feedforward Networks Are Universal Approximatiors," *Neural Networks,* vol. 2, pp. 359-366, 1989.

[16] J. Kleffe, K. Hermann, W. Vahrson, B. Wittig, and V. Brendel, "Logitlinear Models for the Prediction of Splice Sites in Plant Pre-mRNA Sequences," *Nucleic Acids Research,* vol. 24, pp. 4709-4718, 1996.

[17] M. Kozak, "An Analysis of 5'-Noncoding Sequences from 699 Vertebrate Messenger RNAs," *Nucleic Acids Research,* vol. 15, pp. 8125-8148, 1987.

[18] H. Liu and L. Wong, "Data Mining Tools for Biological Sequences," *J. Bioinformatics and Computational Biology,* vol. 1, pp. 139-160, 2003.

[19] J.P. Martens and N. Weymaere, "An Equalized Error Back Propagration Algorithm for the On-Line Training of Multilayer Perceptrons," *IEEE Trans. Neural Networks,* vol. 13, pp. 532-541, 2002.

[20] D. Nguyen and B. Widrow, "Improving the Learning Speed of 2-Layer Neural Networks by Choosing Initial Values of the Adaptive Weights," *Proc. Int'l Joint Conf. Neural Networks,* vol. 3, pp. 21-26, 1990.

[21] U. Ohler, S. Harback, H. Niemann, E. Noth, and G.M. Rubin, "Joint Modeling of DNA Sequence and Physical Properties to Improve Eukaryotic Promoter Recognition," *Bioinformatics,* vol. 17, pp. 199-206, 2001.

[22] U. Ohler, H. Niemann, G. Liao, and M.G. Reese, "Interpolated Markov Chains for Eukaryotic Promoter Recognition," *Bioinformatics,* vol. 15, pp. 362-369, 1999.

[23] D.J. Patterson, K. Yasuhara, and W.L. Ruzzo, "Pre-mRNA Secondary Structure Prediction Aids Splice Site Prediction," *Proc. Pacific Symp. Biocomputing,* pp. 223-234, 2002.

[24] A.G. Pedersen and H. Nielsen, "Neural Network Prediction of Translation Initiation Sites in Eukaryotes: Perspective for EST and Genome Analysis," *Intelligent Systems for Molecular Biology,* vol. 5, pp. 226-233, 1997.

[25] A.G. Pedersen, P. Baldi, Y. Chauvin, and S. Brunak, "The Biology of Eukaryotic Promoter Prediction—A Review," *Computer Chem,* vol. 23, pp. 191-207, 1999.

[26] M. Pertea, L. XiaoYing, and S.L. Salzberg, "GeneSplicer: A New Computational Method for Splice Site Detection," *Nucleic Acids Research,* vol. 29, pp. 1185-1190, 2001.

[27] V.P. Plagianakos, G.D. Magoulas, and M.N. Vrahatis, "Learning Rate Adaptation in Stochastic Gradient Descent," *Advances in Convex Analysis and Global Optimization,* chapter 2, pp. 15-26, 2000.

[28] A. Pinkus, "Approximation Theory of the MLP Model in Neural Networks," *Acta Numerica,* pp. 143-195, 1999.

[29] M.G. Reese, F.H. Eeckman, D. Kulp, and D. Haussler, "Improved Splice Site Detection in Genie," *J. Computational Biology,* vol. 4, pp. 311-324, 1997.

[30] M.G. Reese, "Application of a Time-Delay Neural Network to Promoter Annotation in the Drosophila Melanogaster Genome," *Computer Chem,* vol. 26, pp. 51-56, 2001.

[31] A.A. Salamov, T. Nishikawa, and M.B. Swindells, "Assessing Protein Coding Region Integrity in cDNA Sequencing Projects," *Bioinformatics,* vol. 14, pp. 384-390, 1998.

[32] S.L. Salzberg, A.L. Delcher, K. Fasman, and J. Henderson, "A Decision Tree System for Finding Genes in DNA," *J. Computational Biology,* vol. 5, pp. 667-680, 1998.

[33] M. Scherf, A. Klingenhoff, and T. Werner, "Highly Specific Localization of Promoter Regions in Large Genomic Sequences by PromoterInspector: A Novel Analysis Approach," *J. Molecular Biology,* vol. 297, pp. 599-606, 2000.

[34] S. Sonnenburg, "New Methods for Splice Site Recognition," master's thesis, Humbold Univ., Germany, 2002.

[35] T.A. Thanaraj, "Positional Characterisation of False Positives from Computational Prediction of Human Splice Sites," *Nucleic Acids Research,* vol. 28, pp. 744-754, 2000.

[36] J.T.L. Wang, Q. Ma, D. Shasha, and C.H. Wu, "New Techniques for Extracting Features from Protein Sequences," *IBM Systems J.,* vol. 40, no. 2, pp. 426-441, 2001.

[37] L. Wong, F. Zeng, and R. Yap, "Using Feature Generation and Feature Selection for Accurate Prediction of Translation Initiation Sites," *Proc. Int'l Conf. Genome Informatics,* pp. 192-200, 2002.

[38] M.M. Yin and J.T.L. Wang, "Effective Hidden Markov Models for Detecting Splice Junction Sites in DNA Sequences," *Information Sciences,* vol. 139, pp. 139-163, 2001.

[39] M.Q. Zhang, "Identification of Protein Coding Regions in Human Genome by Quadratic Discriminal Analysis," *Proc. Nat'l Academy of Sciences,* pp. 565-568, 1997.

[40] M.Q. Zhang, "Computational Methods for Promoter Prediction," *Current Topics in Computational Molecular Biology,* chapter 10, pp. 249-267, 2002.

[41] A. Zien, G. Raetsch, S. Mika, B. Schoelkopf, C. Lemmen, A. Smola, T. Lengauer, and K.R. Mueller, "Engineering Support Vector Machine Kernels that Recognize Translation Initiation Sites," *Bioinformatics,* vol. 16, pp. 799-807, 2000.

[42] GeneSplicer, 2005, http://www.tigr.org/tdb/GeneSplicer/gene_spl.html.

[43] GENIO: http://genio.informatik.uni-stuttgart.de/GENIO/splice/, 2005.

[44] HSPL: http://www.softberry.com, 2005.

[45] NNSPlice: http://www.fruitfly.org/seq_tools/splice.html, 2005.

[46] NNSplice Data Set: http://www.fruitfly.org/seq_tools/splice.html, 2005.

[47] PromoterData: http://www.fruitfly.org/seq_tools/datasets/Human/promoter/,2005.

[48] SpliceView: http://l25.itba.mi.cnr.it/webgene/wwwspliceview.html, 2005.

Jagath C. Rajapakse received the BSc (engineering) degree with first class honors from the University of Moratuwa (Sri Lanka) in 1985. He won the award for the best undergraduate in electronic and telecommunication engineering and also received the Fulbright scholarship (1987-1989). He received the MSc and PhD degrees in electrical and computer engineering from the State University of New York at Buffalo in 1989 and 1993, respectively. He was a visiting fellow at the National Institutes of Health (Bethesda) from 1993-1996 and then a visiting scientist at the Max-Planck-Institute of Cognitive Neuroscience (Leipzig, Germany). In May 1998, he joined Nanyang Technological University (Singapore), where he is presently an associate professor in the School of Computer Engineering and also the Deputy Director of the BioInformatics Research Centre (BIRC). He has authored more than 150 research publications in refereed journals, conference proceedings, and books. He also serves on the editorial boards of the journals *Neural Information Processing: Letters and Reviews* and the *International Journal of Computational Intelligence*. He is a senior member of the IEEE, a governing board member of APNNA, and a member of AAAS. His current teaching and research interests are in computational biology, neuroimaging, and machine learning.

Loi Sy Ho received the BS degree with first class honors in computer science from Vietnam National University, Hanoi, in 2000. Since July 2001, he has been a PhD student in the School of Computer Engineering, Nanyang Technological University, Singapore. His research has been in the area of bioinformatics, including computational techniques to detect various signals and structures in biological sequences. He has published more than 10 research papers. Ho has received the Best Student Paper Award and the Overall Best Paper Award at the First IEEE Symposium on Computational Intelligence in Bioinformatics and Computational Biology (CBICB 2004). He was also awarded the Young Scientist Award by the Japanese Society for Bioinformatics for the work he presented at the 14th International Conference on Genomic Informatics (GIW 2003).