Genes Genet. Syst. (2009) **84**, p. 425–430

N4: A precise and highly sensitive promoter predictor using neural network fed by nearest neighbors

Amjad Askary[1,2*], Ali Masoudi-Nejad[1], Roozbeh Sharafi[1], Amir Mizbani[2], Sobhan Naderi Parizi[2] and Malihe Purmasjedi[2]

[1]*Laboratory of Systems Biology and Bioinformatics (LBB), Institute of Biochemistry and Biophysics and COE in Biomathematics, University of Tehran, Tehran, Iran*
[2]*Department of Biotechnology, College of Science, University of Tehran, Tehran, Iran*

(Received 16 July 2009, accepted 26 November 2009)

Promoters, the genomic regions proximal to the transcriptional start sites (TSSs) play pivotal roles in determining the rate of transcription initiation by serving as direct docking platforms for the RNA polymerase II complex. In the post-genomic era, correct gene prediction has become one of the biggest challenges in genome annotation. Species-independent promoter prediction tools could also be useful in meta-genomics, since transcription data will not be available for micro-organisms which are not cultivated. Promoter prediction in prokaryotic genomes presents unique challenges owing to their organizational properties. Several methods have been developed to predict the promoter regions of genomes in prokaryotes, including algorithms for recognition of sequence motifs, artificial neural networks, and algorithms based on genome's structure. However, none of them satisfies both criteria of sensitivity and precision. In this work, we present a modified artificial neural network fed by nearest neighbors based on DNA duplex stability, named N4, which can predict the transcription start sites of *Escherichia coli* with sensitivity and precision both above 94%, better than most of the existed algorithms.

Key words: promoter prediction, neural networks, DNA stability, promoter finder, *E. coli*

INTRODUCTION

In the past decade, bioinformatics has become an integral part of research and development in the biological sciences. Bioinformatics now has an essential role both in deciphering genomic, transcriptomic and proteomic data generated by high-throughput experimental technologies and in organizing information gathered from traditional biology. Thanks to innovations in high-throughput measurement technologies and information technologies, genome-wide analysis is becoming available in a broad range of research fields from DNA sequences, gene prediction, gene expressions, protein structures and interactions, to pathways or networks analysis (Masoudi-Nejad et al., 2007). To date, the genome sequences for over 660 different species have been completely determined and sequencing of 1500 prokaryotic and 854 eukaryotic genomes is ongoing. Although great progresses have been made in gene prediction (Li and Lin., 2006; Wang et al., 2004), one of the most difficult tasks in the annotation of whole genomes is the accurate identification and delineation of promoters (Ohler et al., 1999; Bajic et al., 2004). It can be used for the discovery of genes that are missed by gene predictors and/or genes for which validation tools such as ESTs, cDNAs, etc. are not available. The biggest challenge now is to analyze the acquired sequences, e.g. to locate genes, regulatory sequences, promoters and transcription start sites (TSS). Transcription initiation is the first step in gene expression, and is mainly controlled by transcriptional factors that bind to proximal regions of promoters (Suzuki et al., 2002). The promoter is commonly called as the region upstream of a gene that contains the information necessary for the proper activation or repression of the gene that it controls (Pedersen et al., 1999; Smale and Kadonaga, 2003). The promoter region itself is typically divided into three parts: (1) the core promoter, which is the region that is responsible for the actual binding of the transcription apparatus and is typically situated ~35 bp upstream of the transcription start site (TSS); (2) the proximal promoter, a region containing several regulatory elements, which ranges up to a few hundred base pairs upstream of the TSS; and (3) the distal promoter, which

Edited by Takashi Endo
* Corresponding author. E-mail: LBB@ibb.ut.ac.ir

can range several thousands of base pairs upstream of the TSS and contains additional regulatory elements called enhancers and silencers.

The firstly-identified common feature of prokaryotic promoters was the –10 box with the consensus sequence TATAAT, today known as Pribnow box. By identification of more promoters, more consensus motifs were found, including –35 hexamer TTGACA and some other upstream elements (Seeburg et al., 1977; Hawley and McClure, 1983; Galas et al., 1985; Kumar et al., 1993; Ross et al., 1993; Estrem et al., 1999; Burr et al., 2000). In spite of vast amount of research undertaken for development and improvement of promoter prediction algorithms, currently available prediction methods are not satisfying, mainly due to their poor sensitivity and/or precision. Most of these algorithms search for specific motifs in a sequence to decide whether it is a promoter or not (Staden, 1984; Bucher, 1990; Gordon et al., 2006; Audic and Claverie, 1997), by using techniques such as position weight matrices and Markov models. Artificial neural networks have also been widely employed for prediction of promoters, resulting in highly sensitive prediction methods, though mainly suffering from high frequencies of false positive predictions (Demeler and Zhou, 1991; Knudsen, 1999; Reese and Eeckman, 1995; Yang et al., 1999; Burden et al., 2005; Abeel et al., 2008). Recently, a simple approach has been presented that employs global structural features of the DNA sequence in promoter and non-promoter regions (Benham, 1996). This technique does not use any complex machine learning algorithm, for which it is often impossible to infer any new knowledge from the model itself. This protocol requires no training whatsoever but can be applied only to eukaryotic genomes. Another approach that has been given considerable attention is to use the DNA duplex stability patterns (Wang et al., 2004; Wang and Benham, 2006; Kanhere and Bansal, 2005). The method of Kanhere and Bansal (2005) has been shown to have a better precision compared to previous methods, while having a simple logic that can be easily incorporated into other algorithms such as artificial neural networks.

In this work, we have designed an artificial neural network that obtains a sequence of nucleotides on which slides a 414-nucleotide window with sliding size of one nucleotide. Each window was applied in the form of 413 nearest neighbors (or dinucleotides). Through this article, it is called Neural Network fed by Nearest Neighbors (NNNN or N4). N4 is designed on the top of the algorithm of Kanhere and Bansal (2005). Briefly, the algorithm first computes the average stability over the first 50 and the last 100 nucleotides of a 200nt sequence:

$$E_1 = \sum_{i=1}^{50} \Delta G_i^0 / 50 \quad (1)$$

$$E_2 = \sum_{i=100}^{199} \Delta G_i^0 / 100 \quad (2)$$

where ΔG_i^0 represents the stability of a 15-nt window sequence around the ith nucleotide. This stability is calculated regarding to the nearest neighbor model for prediction of free energy of nucleic acid duplex formation (Breslauer et al., 1986). Since in promoter sequences E_1 is usually larger than E_2, indicating a lower stability upstream of promoter sequences, difference between E_1 and E_2 was used as a predictor parameter:

$$D = E_1 - E_2 \quad (3)$$

The algorithm then computes the value of E_1 and D for each input sequence, and if these values are each larger than a specified threshold, reports a promoter. Sensitivity and precision of the N4 was tested using *Escherichia coli* data. After being trained, N4 was able to predict transcription start sites with the highest sensitivity and precision when compared with other programs.

MATERIALS AND METHODS

The genomic sequence and TSS's of *Escherichia coli* The complete genomic sequence of *Escherichia coli* K12 MG1655 was obtained from EMBL (accession no. U00096). The positions of 467 experimentally determined transcription start sites were retrieved from RegulonDB (Salgado et al., 2004).

The architecture of N4 N4 is designed based on of the algorithm of the DNA stability-based method that was first introduced by Kanhere and Bansal (2005). Figure 1 illustrates the main architecture of N4 (for more

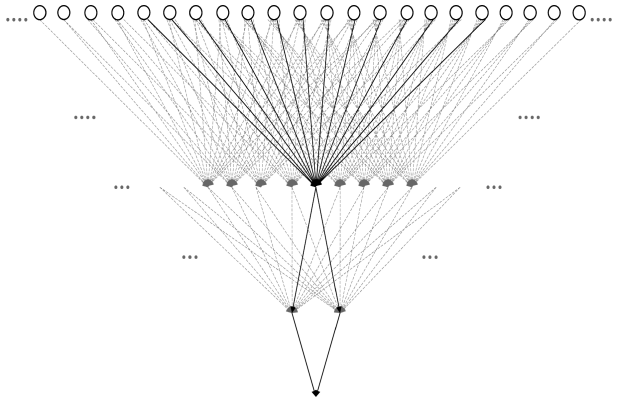


Fig. 1. Architecture of the N4. Each circle shows a group of 16 neurons that can indicate the state of a dinucleotide. Each line in the first row stands for the 16 synapses that a group in the first layer has with a neuron in the second layer. In the other rows, each line stands for one synapse.

information on artificial neural networks, see Laurene, 1994). It is a perceptron network consisting of four layers, which obtains the input sequence as a series of dinucleotides.

The first layer possesses 413 groups each of 16 linear function neurons, adding up to 6608 neurons. Each group represents the state of a dinucleotide and the corresponding input values for the neurons of that group will be a sequence of 16 digits each of which represents existence of energy contribution of a particular dinucleotide to the total DNA duplex stability. Therefore, according to the Fig. 2 all of these digits must be set to zero except the one whose column is identical to the energy contribution of the desired dinucleotide. For example, if the input dinucleotide for a group is "tt", the digit sequence of that group will be {1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0}, and the input pattern for "tc" will be {0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0}. A sequence such as "…ttc…" can be represented by a series of dinucleotides as {…, tt ,tc , …}, and the corresponding input pattern for this sequence will be {…, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, …}. Therefore, N4 obtains a sequence of 414 nucleotides that can be expressed as a series of 413 dinucleotides. The input state of 413 neurons is always 1 and others are 0, depending on the input sequence.

The second layer is composed of 400 sigmoid function neurons, each neuron having a synapse with each of the neurons of the 14 nearest groups of the first layer (adding up to $14 \times 16 = 224$ synapses for each of the second layer neurons). The third layer has two sigmoid function neurons, each having a synapse with each of the neurons of the second layer. Finally, the fourth layer has a single sigmoid neuron that has two synapses with the two neurons of the third layer. An output that is larger than 0.5 is assumed as a TSS.

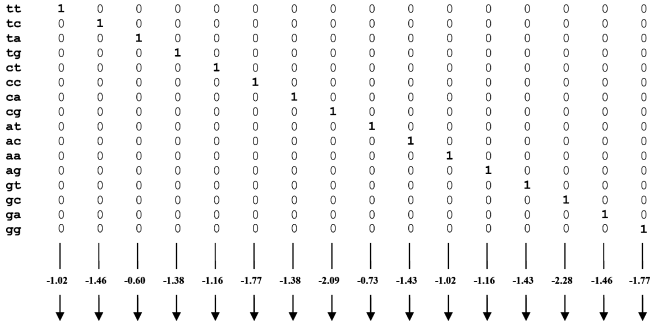


Fig. 2. The initial weightings of the synapses between each group of 16 neurons in the first layer and a neuron in the second layer. The input pattern for each dinucleotide (nearest neighbor) is also indicated. This weighting causes the energy term of each dinucleotide to be passed to the second layer. The local duplex stabilities are then computed in the second layer for windows of 15 nucleotides (14 dinucleotides).

Initial state of N4 As shown in Fig. 2, the synapses between the first layer and the second layer were initially weighted according to the energy contribution of each dinucleotide to the total DNA duplex stability. The synapses between the neurons 101–150 of the second layer and the two neurons of the third layer were weighed as 0.02, while the synapses between the neurons 200–299 of the second layer and the second neuron of the third layer were weighed as –0.01. All other synapses between the second and the third layers were weighed as very small random values, close to zero. The two synapses between the third and the fourth layers were weighted as 0.50. The initial thresholds of the sigmoid functions were set to zero.

Training data We trained N4 with 467 sequences of length 414, each positioning –207…+207 of one of the *E. coli* transcription start sites, as positive dataset. A leave-one-out cross-validation method was used to avoid overtraining. The negative dataset consisted of 414nt sequences of the open reading frames (ORFs) occurring in the first half of *E. coli* genome. The back-propagation algorithm was used to train N4, in which the training dataset consisted of negative sequences, each followed by one of the positive sequences (distributed randomly through the dataset). The training rate of synapses has been set to 0.1 and momentum value was chosen to be 0.9.

Identifying the determinant portions of input sequences After N4 was trained, to determine which portions of input sequences are the most critical ones for recognition of a transcription start site by N4, we randomly altered all nucleotides of each section $i…i+14$ ($1 \le i \le 400$) for each promoter (we defined a promoter as a 414nt sequence containing a TSS exactly in the middle; see RESULTS AND DISCUSSION). Then, we computed the square difference between the original output of N4 for any promoter sequence and the output after alteration of each section. Finally, we averaged the square differences for each section over all promoters that N4 can recognize them as positives. Alternatively, we computed the average amount by which the scores of 1000 random sequences could be increased by optimizing each of the aforementioned sections (each section $i…i+14$ for $1 \le i \le 400$). Furthermore, we used a 414-dimentional hill-climbing algorithm to determine the sequence that would result in the highest score (output) when fed to N4. Note that in each iteration of optimization scenario, there are 414 nucleotides to choose between for optimization.

RESULTS AND DISCUSSION

Evaluating sensitivity and specificity of promoter prediction A measure for the performance of a promoter prediction program is the harmonic mean of the

recall (sensitivity) and the precision (specificity), known as the F-measure (Staden, 1984). The higher this value, the better the program is able to correctly predict promoters. The recall or sensitivity is the number of predicted promoters (TP) divided by the total number of promoters (TP + FN). The precision or specificity is the number of correct predictions (TP) divided by the total number of predictions (TP + FP). After being trained, N4 was able to recognize 438 out of 467 promoters, indicating a sensitivity of about 0.94. We tested N4 over the complete genome of *Escherichia coli* by feeding it with each of the sequences $i \ldots i+413$ from either forward or reverse strand, where i is between 1 and 4639675. As Fig. 3 shows, N4 is not only able to recognize transcription start sites, but also to pinpoint them, i.e. N4 recognizes a TSS only when it is positioned right in the middle of the input sequence.

Overall on forward and reverse strands, N4 returns 1399 false positives, almost evenly distributed between coding sequences and non-coding sequences. This means a frequency of 1 false positive in each 6633 nucleotides - however, it should be mentioned that considering these positives as false positives is debatable, since there is no experimental database of non-promoter sequences. Keeping this ratio, we can estimate that about 29 false positives will be yielded by N4 in 467 non-promoter sequences of length 414nt. Thereafter, precision of N4 was estimated as 0.94.

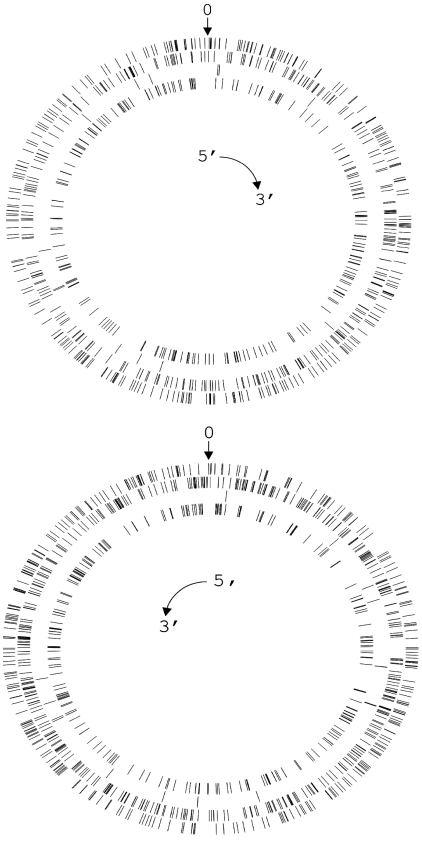
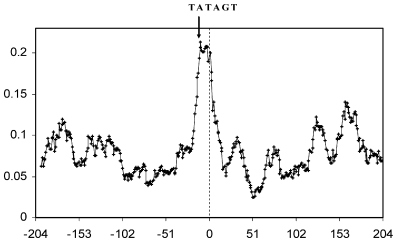


Fig. 3. The distribution of true positives, false positives and false negatives along the genome of *Escherichia coli*. From the outer circle to the inner circle in order: false positives occurring in CDSs, false positives occurring in non-coding regions, false negatives, true positives. **Top:** results for forward strand; **Bottom:** results for reverse strand.

The determinant portions of input sequences Figure 4 (top) shows the average square changes in output of N4 after alteration of 15nt windows of promoter sequences. As it can be discerned, the most informative sequence occurs just before TSS. It is confirmed by the alternative way that was described in the Methods section (Fig. 5). As it was mentioned, a hill-climbing algorithm was used to identify the sequence that would result in the highest score. Figure 4 (bottom) shows the sequence retrieved for the most informative portion. Inter-



ttgtgaacctcttatttgtacctaagt**tatagt**acaagTttttac
-35 -10

Fig. 4. **Top:** The average of square changes in output of N4 after alteration of each 15nt window of promoters. Y-axis shows the average square change of output, while X-axis stands for the position of the middle of the altered window in the promoter sequence. The vertical line shows the TSS. **Bottom:** –38 to +7 of the sequence, which results in the largest output for N4.

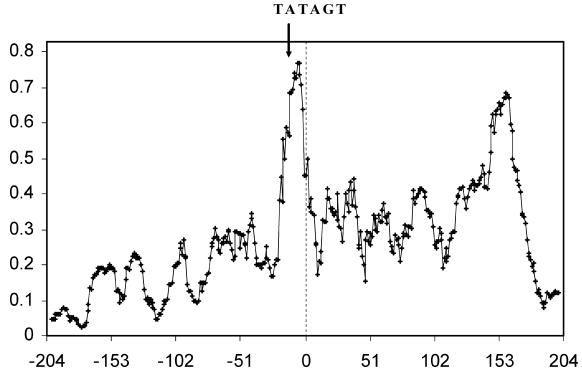


Fig. 5. The average amount by which the N4 output for 1000 random sequences can be increased after optimization of each 15nt window. Y-axis shows the average increase of output, while X-axis stands for the position of the middle of the optimized window in the input sequence. The vertical line shows the mid position (hypothetical TSS).

Table 1. Comparison of **N4** and other promoter prediction programs

Method	Sensitivity	Precision
N4 (this work)	**0.94**	**0.94**
CSVM	0.91	0.85
PCSF	0.91	0.81
Duplex Stability	0.81	0.80
Sequence Alignment Kernel	0.82	0.84
NNPP	0.86	0.54
Weight Matrix of Staden	0.86	0.35

estingly, TATAGT was one of the most important portions of this sequence, which is consistent with the well-known –10 consensus sequence TATAAT. Though with less robustness, however, a consensus sequence could be detected near the position –35, TTGTGA, which is similar to the already known TTGACA.

It can be recognized that 32 out of the 45 nucleotides (71%) in the sequence of Fig. 4 (bottom) are either A or T, indicating a biased nucleotide attendance toward A/T in this region. This is consistent with the results of Wang et al. (2004), which showed that the sequences proximal to transcription start site are less stable than other regions.

Comparison of N4 with other promoter prediction algorithms Table 1 compares the results of N4 with some prominent previously developed methods and some recent methods. As it can be inferred, N4 is superior in terms of both sensitivity and precision. Furthermore, N4 is the only method developed so far that can predict the precise location of transcription start sites with no exception. This high fidelity is due to a combinatory approach that N4 has learned to use.

As we described in the previous sections, the initial values of N4 are based on DNA duplex thermodynamic stability. However, it seems that N4 has also learned to score input sequences regarding sequence motifs that can be found within them, as shown in Fig. 4 (bottom) about motifs –10 and –35. Furthermore, as Fig. 5 shows, there is a portion around the position +160 that can be altered so that N4 would recognize a random sequence as a promoter. Position +160 is downstream of TLS (translation start site) for most transcripts, indicating that N4 probably uses the position of ORF for the accurate prediction of TSS. This approach has already been shown to improve the precision of TSS prediction (Burden et al., 2005).

CONCLUSION

The method by Kanhere and Bansal (2005) was shown to be more reliable than any previously developed method in terms of reducing the frequency of false positives. Therefore, it would be worthy to be employed as an initial state in our algorithm, to be improved by artificial neural network. The architecture and initial state that we chose for N4 made it to perform as much similar to Kanhere and Bansal's original algorithm as possible; hence, the artificial neural network does work better than DNA stability-based method after being trained. Due to the weightings between the first and the second layer (discussed in MATERIALS AND METHODS), each neuron in the second layer receives the duplex stability of a 15nt sequence transmitted through 14 neurons of the first layer as a series of dinucleotides. The first neuron in the third layer receives the average output of 50 neurons of the second layer (neurons 101–150), while the second neuron of the third layer receives the difference between the average output of the previously mentioned 50 neurons and that of 100 downstream neurons (neurons 200–299). These numbers are analogous to the values E_1 and D in Equations 1 and 3. Finally, the neuron in the fourth layer decides whether the outputs of the third layer indicate a promoter or not.

Therefore, N4 was able to reproduce similar results to Kanhere and Bansal's method even before being trained. Training usually makes a network to perform better unless it makes the network overtrained. The leave-one-out cross-validation method that we exploited in this work prevents the latter issue to happen. Thus, it could be expected that N4 algorithm would improve the performance of the Kanhere and Bansal's original algorithm, as results corroborate.

Since Kanhere and Bansal's method was developed for prokaryotic promoter prediction only, and N4 is trained by a set of prokaryotic promoter sequences, it can be assumed as a prokaryotic promoter predictor. However the same approach can be used for development of a eukaryotic promoter predictor on the basis of related thermodynamic principles and experimental data. In this way, learning capability of neural networks can enhance the efficiency of previously established methods.

AVAILABILITY

Source code of the program N4 and some input and training data are freely available upon request from corresponding author (LBB@ibb.ut.ac.ir or amjad.askary@gmail.com).

Authors are grateful to Hamed Shateri and Mehdi Najafsadeghi (Institute of Parasitology, McGill University) for their critical supports.

REFERENCES

Abeel, T., Saeys, Y., Bonnet, E., Rouzé, P., and Van de Peer, Y. (2008) Generic eukaryotic core promoter prediction using structural features of DNA. Genome Res. **18**, 310–323.

Audic, S., and Claverie, J. (1997) Detection of eukaryotic promoters using Markov transition matrices. Comput. Chem. **21**, 223–227.

Bajic, V. B., Tan, S. L., Suzuki, Y., and Sugano, S. (2004) Promoter prediction analysis on the whole human genome. Nat. Biotechnol. **22**, 1467–1473.

Benham, C. J. (1996) Computation of DNA structural variability – A new predictor of DNA regulatory regions. CABIOS **12**, 375–381.

Breslauer, K. J., Frank, R., Blocker, H., and Marky, L. A. (1986) Predicting DNA duplex stability from the base sequence. Proc. Natl. Acad. Sci. USA **83**, 3746–3750.

Bucher, P. (1990) Weight matrix descriptions of four eukaryotic RNA polymerase II promoter elements derived from 502 unrelated promoter sequences. J. Mol. Biol. **212**, 563–578.

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

Burr, T., Mitchell, J., Kolb, A., Minchin, S., and Busby, S. (2000) DNA sequence elements located immediately upstream of the 10 hexamer in *Escherichia coli* promoters, a systematic study. Nucleic Acids Res. **28**, 1864–1870.

Demeler, B., and Zhou, G. (1991) Neural network optimization for *E. coli* promoter prediction. Nucleic Acids Res. **19**, 1593–1599.

Estrem, S. T., Ross, W., Gaal, T., Chen, Z. W., Niu, W., Ebright, R. H., and Gourse, R. L. (1999) Bacterial promoter architecture, subsite structure of UP elements and interactions with the carboxy-terminal domain of the RNA polymerase alpha subunit. Genes Dev. **13**, 2134–2147.

Galas, D. J., Eggert, M., and Waterman, M. S. (1985) Rigorous pattern recognition methods for DNA sequences. Analysis of promoter sequences from *Escherichia coli*. J. Mol. Biol. **186**, 117–128.

Gordon, J. J., Towsey, M. W., Hogan, J. M., Mathews, S. A., and Timms, P. (2006) Improved prediction of bacterial transcription start sites. Bioinformatics **22**, 142–148.

Hawley, D. K., and McClure, W. R. (1983) Compilation and analysis of *Escherichia coli* promoter DNA sequences. Nucleic Acids Res. **11**, 2237–2255.

Kanhere, A., and Bansal, M. (2005) A novel method for prokaryotic promoter prediction based on DNA stability. BMC Bioinformatics **6**, 1–10.

Knudsen, S. (1999) Promoter2.0, for the recognition of polII promoter sequences. Bioinformatics **15**, 356–361.

Kumar, A., Malloch, R. A., Fujita, N., Smillie, D. A., Ishihama, A., and Hayward, R. S. (1993) The minus 35-recognition region of *Escherichia coli* sigma 70 is inessential for initiation of transcription at an "Extended minus 10" promoter. J. Mol. Biol. **232**, 406–418.

Fausett, L. (1994) Fundamentals of Neural Network: Architectures, Algorithms and Applications. pp. 461. Prentice Hall; Englewood Cliffs.

Li, Q. Z., and Lin, H. (2006) The recognition and prediction of SIGMA 70 promoters in *Escherichia coli* K-12. J. Theor. Biol. **242**, 135–141.

Masoudi-Nejad, A., Goto, S., Jauregui, R., Ito, M., Kawashima, S., Moriya, Y., Endo, T. R., and Kanehisa, M. (2007) EGENES, transcriptome-based plant database of genes with metabolic pathway information and expressed sequence tag indices in KEGG. Plant Physiol. **144**, 857–866.

Ohler, U., Harbeck, S., Niemann, H., Noth, E., and Reese, M. (1999) Interpolated Markov chains for eukaryotic promoter recognition. Bioinformatics **15**, 362–369.

Pedersen, A. G., Baldi, P., Chauvin, Y., and Brunak, S. O. (1999) The biology of eukaryotic promoter prediction—A review. Comput. Chem. **23**, 191–207.

Reese, M. G., and Eeckman, F. H. (1995) Novel Neural Network Prediction Systems for Human Promoters and Splice Sites. In Proceedings of the Workshop on Gene-Finding and Gene Structure Prediction. Pennsylvania, Philadelphia, edited by D. Searls, J. Fickett, G. Stormo and M. Noordewier.

Ross, W., Gosink, K. K., Salomon, J., Igarashi, K., Zou, C., Ishihama, A., et al. (1993) A third recognition element in bacterial promoters, DNA binding by the alpha subunit of RNA polymerase. Science **262**, 1407–1413.

Salgado, H., Gama-Castro, S., Martinez-Antonio, A., Diaz-Peredo, E., Sanchez-Solano, F., Peralta-Gil, M., Garcia-Alonso, D., Jimenez-Jacinto, V., Santos-Zavaleta, A., Bonavides-Martinez, C., and Collado-Vides, J. (2004) RegulonDB (version 4.0), transcriptional regulation, operon organization and growth conditions in *Escherichia coli* K-12. Nucleic Acids Res. **32** (Database issue), D303–D306.

Seeburg, P. H., Nusslein, C., and Schaller, H. (1977) Interaction of RNA polymerase with promoters from bacteriophage. Eur. J. Biochem. **74**, 107–113.

Smale, S. T., and Kadonaga, J. T. (2003) The RNA polymerase II core promoter. Annu. Rev. Biochem. **72**, 449–479.

Staden, R. (1984) Computer methods to locate signals in nucleic acid sequences. Nucleic Acids Res. **12**, 505–519.

Suzuki, Y., Yamashita, R., Nakai, K., and Sugano, S. (2002) DBTSS, database of human transcriptional start sites and full-length cDNAs. Nucleic Acids Res. **30**, 328–331.

Wang, H., and Benham, C. J. (2006) Promoter prediction and annotation of microbial genomes based on DNA sequence and structural responses to superhelical stress. BMC Bioinformatics **7**, 248–262.

Wang, H. Q., Noordewier, M., and Benham, C. J. (2004) Stress-induced DNA duplex destabilization (SIDD) in the *Escherichia coli* genome, SIDD sites are closely associated with promoters. Genome Res. **14**, 1575–1584.

Yang, J., Parekh, R., Honavar, V., and Dobbs, D. (1999) Data-driven theory refinement algorithms for bioinformatics. In Proceedings of the International Joint Conference on Neural Networks (IJCNN '99), Washington, DC. IEEE **6**, 4064–4068.