An Intelligent System for Vertebrate Promoter Recognition

Vladimir B. Bajic, Allen Chong, Seng Hong Seah, and Vladimir Brusic,
Laboratories for Information Technology

Dragon Promoter Finder, an advanced system for promoter recognition in vertebrates, uses a collection of models based on multisensor integration, signal processing, and artificial neural networks. The system minimizes the number of false positive predictions for various prespecified sensitivity levels and exhibits several times higher accuracy than other compared systems.

An important task that molecular biology faces today is the characterization and annotation—recognition and cataloging—of genes from various genomes. It contributes to our general knowledge and understanding of the functioning of various organisms' genetic makeup. Bioinformatics gene search methods either are based on homology analysis of the potential gene product or on the analysis of signals that indicate gene presence. One of the latest gene search methods uses *promoter* recognition.[1] A promoter, the DNA region encompassing a gene's *transcription start site* (TSS), largely controls the biological activation of the gene.[2] Promoters contain docking sites for specialized proteins that synergistically initiate gene transcription. One gene has at least one promoter region, although one promoter can participate in regulating several genes' transcription. In eukaryotes, a promoter usually appears near the beginning of the gene whose transcription it regulates. Moreover, promoter recognition allows searching for specific groups of transcriptionally coregulated genes (see the "Promoter Recognition" sidebar). Although a class of coregulated genes might have a similar regulation pattern, their products might not share any homology.

Until recently, efficient gene hunting using promoter recognition was impossible. All techniques developed for promoter recognition for any level of *true positive* recognition also produced a high number of *false positive* recognition.[2] In a TP recognition, the predictor correctly indicates a promoter's presence, and in an FP recognition, the predictor indicates a promoter's presence where a promoter does not exist.

However, the PromoterInspector[3] system allows a promoter-based gene search and produces considerably reduced levels of FPs compared to other publicly available promoter recognition programs. Motivated by this method of gene hunting, we developed a new system, the Dragon Promoter Finder (DPF, http://sdmc.krdl.org.sg/promoter), for general promoter finding. It combines a novel, nonlinear promoter recognition model, signal processing, artificial neural networks (ANNs), and newly developed sensors. We based the sensors on the statistical concept of oligonucleotide positional distributions in specific functional regions of DNA and modeled these distributions as a set of position weight matrices of the most significant oligonucleotides. We evaluated DPF version 1.2 on a sequence-set containing 146 human and human-viral DNA sequences. DPF appears to be several times more accurate than the best publicly available general promoter recognition systems,[3–5] including PromoterInspector.

Dragon Promoter Finder

Figure 1 shows DPF's conceptual structure. The overall model is a composite collection of individual models that possess identical structures. We trained each individual model for a narrow specificity range. Users can request a specific accuracy from the list provided to activate the corresponding model. Data processing in each model is analogous. A data window slides along the DNA sequence, each time shifting one bp ahead (see the "Notation" sidebar). The recognition system analyzes the data window's content. First, the data passes through three parallel sensors. Each sensor models a particular functional region of a gene: promoter, coding-exon, and intron. The system further processes these sensors' outputs and feeds them into an ANN, which performs multisensor integration. We trained each model to separate promoter from nonpromoter

1094-7167/02/$17.00 © 2002 IEEE

Promoter Recognition

A DNA molecule is composed of two complementary strands consisting of four bases: adenine, cytosine, guanine, and thymine (A, C, G, and T). Genes are functional segments of DNA, encoding mainly protein products in a cell. Different genes get activated under different specific physiological conditions. Transcription is the first step in protein production, in which an enzyme RNA polymerase must bind to a gene's promoter region to initiate its transcription. We focus on genes transcribed by RNA polymerase II. This class of eukaryotic genes contains all genes known to code for proteins. Polymerase II cannot directly bind to the promoter region; it requires numerous other proteins (called general *transcription factors*, or TFs) to first bind at transcription factor binding sites (TFBSs). The TFs together with RNA polymerase II form a transcription preinitiation complex, which starts the transcription process. The *transcription start site* (TSS) is where transcription starts on DNA. You can find more detailed introductions to the role of promoters in transcriptional regulation elsewhere.[1,2]

One approach for promoter recognition attempts to recognize different TFBSs. However, numerous problems exist with this:

- You can associate a huge number of potential TFBSs with a promoter, but only a handful play a regulatory role.
- TFBSs can appear in different combinations on different promoters.
- The order of TFBSs in promoters varies.
- Relative distances of TFBSs in various promoters differ.

Eukaryotic promoter structures in larger promoter groups do not share many common features, making promoter recognition difficult with TFBS recognition. However, this approach mimics the inherent biological regulatory mechanism of promoter regions. You can find typical examples of systems that use such techniques elsewhere.[3,4]

Another data-driven approach attempts to determine statistical regularities of promoter and other functional sections of DNA. This statistical approach does not pinpoint the TSS location as precisely as the TFBS signal recognition approach. However, this approach allows for recognition of broader promoter groups. PromoterInspector and Dragon Promoter Finder are based on this approach. Additional approaches are discussed elsewhere.[2]

References

1. A.G. Pedersen et al., "The Biology of Eukaryotic Promoter Prediction: A Review," *Computers & Chemistry*, vol. 23, nos. 3–4, June 1999, pp. 191–207.
2. J.W. Fickett and A.G. Hatzigeorgiou, "Eukaryotic Promoter Recognition," *Genome Research*, vol. 7, no. 9, Sept. 1997, pp. 861–878.
3. S. Knudsen, "Promoter2.0: For the Recognition of Pol II Promoter Sequences," *Bioinformatics*, vol. 15, no. 5, May 1999, pp. 356–361; www.cbs.dtu.dk/services/Promoter.
4. M.G. Reese, N.L. Harris, and F.H. Eeckman, "Large-Scale Sequencing Specific Neural Networks for Promoter and Splice Site Recognition," *Proc. Pacific Symp.* (Biocomputing 96), World Scientific Publishing, Singapore, 1996; pp. 737–738; www.fruitfly.org/seq_tools/promoter.html.

regions. We consider all scores that make the ANN output greater than the selected threshold to be positive predictions in the promoter region.

We derived models of a gene's functional regions as positional distributions of overlapping *pentamers* (all sequences of five consecutive nucleotides) in a region. We used only those pentamers that most significantly contribute to the separation between the promoter and nonpromoter regions. We determined the pentamers' significance using their statistical relevance. For each of the 1,024 possible pentamers p_j, we calculated the relevance function as $J = (\mu_p - \mu_n)/(\tau_p + \tau_n + 1)$, where μ_p was the percentage of promoters in which p_j appeared and μ_n was the percentage of nonpromoters where p_j appeared. The numbers τ_p and τ_n represent p_j's average number of occurrences in sequences in which p_j appears in promoters and nonpromoters, respectively. We ranked pentamers according to the relevance function value and selected the highest 256 for inclusion in the model. Selected pentamers'

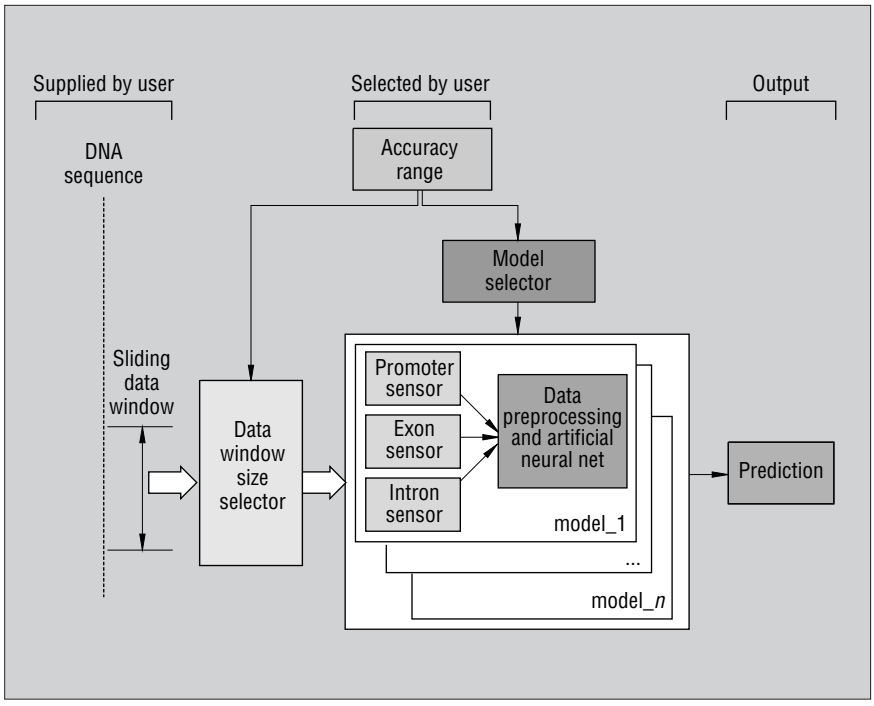

Figure 1. Dragon Promoter Finder's overall structure.

Notation

ANN:	artificial neural network
bp:	base pair
Mbp:	one million bp
DNA:	deoxyribonucleic acid
EPD:	Eukaryotic Promoter Database
FP:	false positive
PWM:	position weight matrix
RNA:	ribonucleic acid
SPB:	signal processing block
TFBS:	transcription factor binding site
TP:	true positive
TSS:	transcription start site

positional distributions were represented by their *positional weight matrices*. We generated the PWMs from the training set for each of the three functional groups, by counting frequencies of all selected pentamers at each position. The PWM of overlapping selected pentamers has dimensions 256 x $(L - 4)$ for a data window of length L. The "very high" specificity model used $L = 250$ nucleotides, and all other models used $L = 200$ nucleotides. This data window slid along the sequence one nucleotide at a time. We compared the data window's content to the weight matrix to calculate the content's representational score:

A data window is considered as containing the sequence $W = n_1 n_2 \ldots n_{L-1} n_L$, where $n_j \in \{A, C, G, T\}$ are nucleotides from the DNA sequence. The corresponding sequence P of successive overlapping pentamers p_j obtained from this data window W is $P = p_1 p_2 \ldots p_{L-5} p_{L-4}$. The following formula produces the score for each data window:

$$S = \frac{\left(\sum_{i=1}^{L-4} p_j^i \otimes f_{j,i} \right)}{\left(\sum_{i=1}^{L-4} \max_j f_{j,i} \right)},$$

$$p_j^i \otimes f_{j,i} = \begin{cases} f_{j,i}, & \text{if } p_i = p_j^i \\ 0, & \text{if } p_i \neq p_j^i \end{cases},$$

where $p^i{}_j$ is the jth pentamer at position i, and $f_{i,j}$ is the frequency of the jth pentamer at position i. These scores take values between 0 and 1. We worked assuming that the higher the score value, the more likely the data window represents the respective functional region. We denoted the scores (signal values) of the promoter, coding-exon, and intron sensors as σ_p, σ_e, and σ_i. We used these scores as inputs to the nonlinear *signal processing block*. The inputs to the SPB entered the nonlinear block, producing three signals z_E, z_I, and z_{EI} that we defined as

$$z_E = blin(\sigma_p - \sigma_e, a_e, b_e, c_e, d_e),$$
$$z_I = \text{blin}(\sigma_p - \sigma_i, a_i, b_i, c_i, d_i),$$
$$z_{EI} = \text{blin}(\sigma_e - \sigma_i, a_{ei}, b_{ei}, c_{ei}, d_{ei}),$$

where the function *blin* was defined by

$$blin(x, a, b, c, d) = \begin{cases} cx, & \text{if } x > a \\ x, & \text{if } b \leq x \leq a. \\ dx, & \text{if } b > x \end{cases}$$

Parameters a_k, b_k, c_k, d_k, $k = e, i, ei$, are part of the system's tuning parameters. Then, we normalized the signals z_E, z_I, and z_{EI} by whitening, producing three signals s_E, s_I, and s_{EI} as the SPB's output. We fed these signals as inputs to the ANN system. The threshold that best separated promoters from nonpromoter sequences on the model's tuning set was used. We considered that all ANN outputs greater than the threshold indicate both the promoter region's presence in the data window and the TSS at a position 50 bp before the data window's end. Each submodel's ANN was a simple feed-forward network combined with the nonlinear SPB. The Bayesian regularization method trained the ANNs for the best separation between classes of input signals.

DPF training and testing

We compared the DPF with three general promoter recognition programs (see the "Compared Programs" sidebar). We selected these because of their Web accessibility and because they can analyze long sequences. Promoter 2.0[4] and NNPP 2.1 (Neural Network Promoter Prediction)[5] use ANNs. PromoterInspector (www.genomatix.de) doesn't use ANNs but was reported to significantly outperform five other promoter recognition systems.[3] We thus felt it represented the most efficient stand-alone promoter recognition system.

Training set

DPF was trained on a collection of promoter and nonpromoter sequences. We obtained the promoter sequences from the Eukaryotic Promoter Database.[6] We used 793 different vertebrate promoter sequences of length 250 bp contained in EPD Release 65 and covering a region of 200 bp for very high specificity models (150 bp for other models) before and 50 bp after the TSS, denoted by [–200, +50]. These 250-bp sequences represent positive training data. We also collected a set of nonoverlapping human coding-exon and intron sequences, 250 bp each, from the Genebank Release 121.[7] In total, we used 800 exon and 4,000 intron sequences.

Tuning set

To tune the adjustable system parameters—such as the sensor signals' bounds or threshold levels for sensors and ANNs—we extended the training set by 400 nonoverlapping sequences from the so-called 3' untranslated regions (3'UTR) of humans, 200 new nonoverlapping human coding-exon sequences, and 500 new nonoverlapping human intron sequences (each sequence was 250 bp long). Additionally, we added 20 gene sequences of full sequence length with known TSSs, where the TSSs were not overlapping with the EPD data. We used this extended data set as the tuning set because

- We did not have enough diverse promoter sequences to make two large, separate sets—one for training and one for tuning
- Adding the 3'UTR sequences and new exon and intron sequences extended the negative data
- Adding 20 full-length gene sequences provided a more realistic operational environment for the final tuned system

Tuning is aimed at minimizing the number of FPs for the preselected sensitivity levels. We built different models, each tuned separately for a preselected sensitivity. For the higher specificity range (0.8 to 1), we used a data window of 250 bp. For the lower specificity ranges (0.2 to 0.7), we used a data window of 200 bp because our previous experiments showed that with this window length we could achieve higher sensitivity (0.66) than with the 250-bp data window. We selected five different models for DPF's public version.

Test set A

We compiled evaluation set A from a larger set of human and human-viral sequences that

Compared Programs

PromoterInspector is a region-predicting general promoter recognition program. It uses models of four functional regions of genes: promoter, exon, intron, and 3'UTR. It derives models as generalized IUPAC (International Union of Pure and Applied Chemistry) groups of region characteristic oligomers[1] obtained from 100-bp sequence segments (see the "Notation" sidebar). For promoter modeling, Matthias Scherf and his colleagues derived these segments from the EPD data, using regions of [–500, +50] relative to the TSS. For nonpromoter models, they derived 100-bp segments from sequences collected randomly from the GenBank database (totaling 1 Mbp for each nonpromoter group).[1] The system uses a data window of 100 bp that slides along the DNA strand, shifting 4 bp ahead each time. The four region sensors compete, and the promoter sensor signal must be stronger than the other three sensors' signals. PromoterInspector predicts a promoter on the occurrence of a minimum of 24 successive positive predictions.

Promoter 2.0 is based on ANNs and trained to recognize four specific signals most commonly present in eukaryotic promoters—TATA box, initiator (Inr), GC-box, and CCAAT-box—as well as their mutual distances. However, DPF makes 27 times fewer FP recognitions than this system with the same level of TP recognition.

NNPP 2.1 is based on the recognition of two specific signals within the promoter region—the TATA box and the initiator—as well as their mutual distances. This system uses three time-delay ANNs, one for recognition of the TATA box, one for the Inr, and one that combines the outputs of the two and accounts for the spatial distance between these signals. NNPP 2.1 has been trained on promoter data from the EPD and nonpromoter data from the Genie training set. Although set A contains a significant portion of sequences from Genie training set, DPF outperforms NNPP 2.1 by roughly 3 to 5.3 times in accuracy.

Reference

1. M. Scherf, A. Klingenhoff, and T. Werner, "Highly Specific Localisation of Promoter Regions in Large Genomic Sequences by PromoterInspector: A Novel Context Analysis Approach," *J. Molecular Biology*, vol. 297, no. 3, Mar. 2000, pp. 599–606.

other researchers compiled and used in various gene recognition and promoter-finding projects. We used these criteria for including sequences in the evaluation set:

- The sequence should be from a relatively large collection of human sequences (not fewer than 20) used previously either to evaluate promoter recognition systems or to train and evaluate gene recognition systems. This reduces possible bias in sequence selection, preserves the diversity of the promoter regions used, and ensures a more representative evaluation set. General gene recognition programs are trained to recognize broad classes of genes. So, they are trained on diverse sets of gene sequences that are excellent candidates for evaluating promoter recognition systems. We included sequences used to train the gene-finding and analysis programs Genie,[8] Genescan, and NetGene and sequences used to test, among other things, a promoter recognition program.[9] The sequences used for training Genie constituted most of our evaluation set.
- No sequence in the evaluation set can have any part used for either training or tuning our system. This condition eliminated all sequences in the EPD's current release and other sequences whose exon, intron, or 3'UTR parts were used for DPF's training or tuning.
- The sequences should have a sufficiently detailed and complete annotation of the gene's 5' flanking region, enabling identification of the TSS location and, therefore, promoter location.
- Finally, we checked the sequences that satisfied the above three criteria against the literature and GenBank annotation. We excluded those with ambiguous or conflicting annotation regarding the gene start and possible promoter location.

The final evaluation set contained 159 TSSs, 146 human and human-viral sequences, and a cumulative length of more than 1.1 Mbp. Table 1 summarizes the set's contents.

Test set B

We also evaluated DPF performance on test set B,[3] first introduced to assess the performance of PromoterInspector and several other promoter recognition programs. This set contains six sequences with 35 TSSs and has a length of approximately 1.38 Mbp. This set's analysis indicates biases, so you should interpret the results with caution.

First, 26 of 35 (74 percent) promoters in this set are CpG island-related. CpG islands are unmethylated DNA segments longer than 200 bp, have at least 50 percent C+G content, and have at least 60 percent the number of CpG dinucleotides that you might expect from the segment's C+G content; CpG islands are found around TSSs in approximately one-half of the vertebrate promoters.[2]

Second, only five promoters have the rel-

Table 1. The composition of evaluation set A.

Sets	Number of sequences	Number of TSSs	Total length (bp)
Subset from Niels Mache and colleagues[10]	29	29	175,436
Subset of the Genie training set[9]	97	98	873,563
Subset from other gene recognition programs (see http://sdmc.lit.org.sg/promoter/promoter1_2/DPFV12.htm)	20	32	103,909
The whole evaluation set	146	159	1,152,908

Table 2. Promoter characteristics in set A and set B.

Promoter Test Sets	Set A	Set B
CpG island-related promoters	90 (56.6%)	26 (74.3%)
TATA box in correct context	106 (66.7%)	5 (14.3%)

Table 3. Results on evaluation set A with matched sensitivity levels. The results for NNPP 2.1 are given for two thresholds.*

Program	Se	#TP	#FP	#FP/#FPdpf
DPF	0.22	35	35	
PromoterInspector	0.22	35	117	3.36
DPF	0.25	41	64	
Promoter 2.0	0.25	41	1,764	27.56
DPF	0.28	45	78	
NNPP 2.1 (th = 0.99)	0.28	45	415	5.32
DPF	0.66	106	994	
NNPP 2.1 (th = 0.8)	0.66	106	3,070	3.08

* Se is the sensitivity; #TP is the number of correct predictions; #FP is the number of false positive predictions. #FPdpf is the number of false positive predictions made by DPF. #FP/#FPdpf shows the fold reduction of DPF false positive predictions compared with other programs.

atively common TATA box element in the correct promoter context. So, promoter prediction programs such as Promoter 2.0, NNPP 2.1, TSSG, and TSSW,[3] in which a TATA box sensor forms a component of their recognition systems, perform poorly on set B. This also partly explains the poor performance of several other programs on set B and shows that set B might not be a good choice for the assessment of promoter prediction programs. Table 2 summarizes the characteristics of sets A and B.

Human chromosome 22

Finally, we evaluated DPF's performance on Release 2.3 of the human chromosome 22 and annotation data produced by the Chromosome 22 Gene Annotation Group at the Sanger Institute (www.sanger.ac.uk/HGP/Chr22). Chromosome 22 has a much higher C+G content than most other human chromosomes. Approximately 65 percent of the 339 annotated known genes are CpG island-related.

Results

We illustrate our system's performance against other promoter-prediction programs in the following sections.

Set A

The first of the four criteria used in creating set A, as we discussed earlier, helped minimize the unintentional bias in the dataset's composition. Moreover, Anders Pedersen and his colleagues estimate that about one-half of all vertebrate promoters are CpG island-related;[2] also, approximately 70 percent of promoters have a TATA box element. Set A's composition roughly resembles these estimates. Table 3 summarizes the evaluation results on set A for the compared programs.

Set A hits criteria

We adopted the criterion for assigning correct or incorrect prediction from James Fickett and Artemis Hatzigeorgiou's work.[10] We considered the predicted TSS correct if it fell within 200 nucleotides upstream or 100 nucleotides downstream of the real (experimentally determined) TSS. We used this criterion for assessing programs that predict TSSs as "pinpointed" locations, such as DPF, Promoter 2.0, and NNPP 2.1. NNPP 2.1 gives the region around the predicted TSS location in the range [–40, +10], so that you can take the TSS location as 40 bp downstream of the region's predicted 5' boundary.

For PromoterInspector, which predicts the promoter as a region, we considered the prediction correct if the region contained a real TSS. Otherwise, we considered the prediction false. We based this on the generally accepted concept that a promoter must contain a TSS. Moreover, the average length of the promoter regions predicted by PromoterInspector is 420 bp on set A. So, the prediction range considered by PromoterInspector as a TP guess is approximately 40 percent greater than the 300 bp (200 bp upstream and 100 bp downstream of the real TSS) boundary criterion afforded to all other prediction programs studied here, including DPF.

The DNA molecule exists in vivo as a double-stranded molecule, and a gene can be found on either the so-called *positive* or *negative* strand. Predictions of NNPP 2.1, Promoter 2.0, and DPF are strand-specific, so they can identify the gene's start on the respective strand. PromoterInspector identifies only the promoter's general location on the genomic sequence, disregarding the strand orientation. In PromoterInspector,[3] TPs are counted as correct irrespective of the strand on which a real TSS is found, which is different from the way we counted it. You should really consider each prediction that PromoterInspector makes as two predictions (one for each strand) to assess the prediction program's performance. For the same sensitivity, DPF produced three to 26 times less FP predictions than the compared programs (see Table 3).

Set B

Table 2 reflects set B's compositional features, which poorly resemble the general characteristics of vertebrate promoters. However, you can use it to assess the prediction ability of promoters that are CpG island-related and have no TATA box elements. Because of these limitations, we can't draw conclusions about any program's ability to find general promoters based on the results obtained on this set. Tables 4a and 4b summarize the achieved results. The evaluation results on set B indicate that DPF generalizes well because its prediction accuracy is consistent with those on other evaluation sets.

Set B hits criteria

To compare PromoterInspector and DPF, we considered a DPF prediction as correct if a predicted TSS fell within an interval of 500 bp (average length of the promoter region predicted by PromoterInspector on set B), with all other conditions the same as those used for PromoterInspector.[3] The number of FP predictions on set B made by PromoterInspector is 54 by our criteria because strand-nonspecific predictions were counted twice (once for each DNA strand). The different criteria Matthias Scherf and his colleagues used for PromoterInspector and for TSS-finding programs[3] prevent direct comparison of the

Table 4. Results on set B: (a) NNPP 2.1, Promoter 2.0, and PromoterInspector results and (b) DPF results. #TP is the number of correct predictions; #FP is the number of false positive predictions.

Set B performance	NNPP 2.1	Promoter 2.0	PromoterInspector
#TP	23	8	16
#FP	3,533	1,751	54

(a)

Set B performance	DPF Se = 0.22	DPF Se = 0.30	DPF Se = 0.37	DPF Se = 0.40	DPF Se = 0.50	DPF Se = 0.66
#TP	8	9	16	19	20	27
#FP	22	53	79	144	227	543

(b)

reported results[3] of NNPP 2.1 and Promoter 2.0 (used in Tables 4a and 4b). However, we present them for the sake of completeness.

Human chromosome 22

To show the performance of DPF on larger genomic contigs, we also tested its behavior on human chromosome 22.

Chromosome 22 hits criteria

For PromoterInspector, TP predictions were determined according to criteria detailed elsewhere,[1] and FP predictions were those that fell along the length of a known gene but were not counted as TP predictions. For DPF, we counted the predicted TSS position as correct if it fell within a length interval equal to the average length of the promoter region predicted by PromoterInspector on human chromosome 22 (555 bp), with all other conditions the same as those used for PromoterInspector.[1] This makes the criteria for comparison between the two programs equivalent. Table 5 compares the performances on set A and human chromosome 22.

CpG island-related promoters and DPF's performance

DPF recognizes well CpG island-related (*CpG+*) and nonrelated (*CpG–*) promoters. *#CpG+* denotes the number of TPs that recognize *CpG+* promoters, and *#CpG–* denotes the number of TPs that recognize *CpG–* promoters. The ratio of *#CpG–* to *#CpG+* promoters in set A is 0.7667. Table 6 summarizes the results of DPF for set A, relative to CpG island-related promoters.

DPF is a general TSS-finding program, not specialized to particular vertebrate promoter groups. It can successfully recognize both CpG island-related and CpG island-nonrelated promoters. Its performance on several large sets (A, B, and human chromosome 22) is reasonably consistent even though these sets have a different composition of promoter types, and it outperforms other TSS-finding programs. It achieves sensitivities of over 77 percent on set B, but on average, its expected maximum sensitivity is approximately 66 percent. In general, the DPF produces many times fewer FP predictions than comparative systems at the same sensitivity level. Approximately 30 percent of TP predictions are within several bp of the real TSS, another 30 percent are 20 to 50 bp upstream of the real TSS, 30 percent are within 50 to 150 bp shifted upstream of the real TSS, and 10 percent are downstream (data not shown).

PromoterInspector is a program with a considerably improved #TP/#FP ratio compared to several other promoter recognition programs.[3] Although PromoterInspector is a general promoter finding program, its performance varies on the different test sets. It achieves a sensitivity of approximately 0.46 on set B. 56.6 percent of promoters in set A are CpG island-related. PromoterInspector predicts 39 percent of CpG island-related promoters in set A: This represents 22 percent of all promoters in set A, which, coincidently, is PromoterInspector's sensitivity on set A. The performance of each promoter prediction program (DPF, PromoterInspector, and so on) will vary to some extent on different test sets depending on these sets' compositional characteristics.

Table 5. The performance Comparison on set A and human chromosome 22. #TP is the number of correct predictions; #FP is the number of false positive predictions.

Program	Scores	set A	Human chromosome 22 (known genes)
DPF, Se = 0.22	TP (%) #FP/#TP	22 1	20.06 0.75
DPF, Se = 0.3	TP (%) #FP/#TP	30 2.06	30.67 2.798
DPF, Se = 0.4	TP (%) #FP/#TP	40 3.64	60.177 3.5637
PromoterInspector	TP (%) #FP/#TP	22 3.3439	45 1.975

Table 6. DPF predictions of CpG island-related (*CpG+*) and nonrelated (*CpG–*) promoters from set A. The last row indicates the ratio of recognized CpG-nonrelated to CpG-related promoters.

Sensitivity	66 %	50 %	40 %	30 %	22 %
#CpG+	67	47	33	24	19
#CpG–	39	32	31	25	16
Total TP	106	79	64	48	35
#CpG–/ #CpG+	0.5821	0.6809	0.9391	1.0417	0.8421

Vladimir B. Bajic is the deputy head of the BioDiscovery Group at the Laboratories for Information Technology in Singapore and an adjunct full professor of bioinformatics at the South African National Bioinformatics Institute (SANBI). His research interests include applications of artificial intelligence in bioinformatics. He holds a DEngSc in electrical engineering from the University of Zagreb, Yugoslavia. Contact him at Laboratories for Information Technology, 21 Heng Mui Keng Terrace, Singapore 119613; bajicv@lit.a-star.edu.sg.

Allen Chong is a researcher in the BioDiscovery Group at the Laboratories for Information Technology in Singapore. His research interests are in neurophysiology, particularly in the field of neurodegenerative diseases. He holds a BS in physiology from University College, London. Contact him at Laboratories for Information Technology, 21 Heng Mui Keng Terrace, Singapore 119613; achong@lit.a-star.edu.sg.

Seng Hong Seah is an engineer in the BioDiscovery Group at the Laboratories for Information Technology in Singapore. His research interests include software engineering and programming in bioinformatics. He holds a BS in computer science from Nanyang Technological University, Singapore. Contact him at Laboratories for Information Technology, 21 Heng Mui Keng Terrace, Singapore 119613; shseah@lit.a-star.edu.sg.

Vladimir Brusic is the head of the BioDiscovery Group at the Laboratories for Information Technology in Singapore. His research interests include modeling complex biological systems and knowledge discovery from databases. He has a PhD in bioinformatics from La Trobe University, Australia, and master's degrees in biomedical engineering (University of Belgrade, Yugoslavia), information technology (Royal Melbourne Institute of Technology, Australia), and business administration (Rutgers University). Contact him at Laboratories for Information Technology, 21 Heng Mui Keng Terrace, Singapore 119613; vladimir@lit.a-star.edu.sg.

On set A, DPF makes 3.34 times fewer FP predictions than PromoterInspector while making the same number of TPs. Furthermore, the consistency of DPF's performance on all data sets used here demonstrates that the model we used provides reliable identification of a wider promoter group and does not favor a specific promoter type (such as CpG island-related promoters). PromoterInspector's performance supports our observation that it predicts different classes of promoters, although it favors CpG island-related ones. Theoretically, owing to nonstrand-specific predictions, PromoterInspector cannot achieve a positive predictive value greater than 0.5, because it produces one FP prediction for each TP prediction. Furthermore, PromoterInspector cannot pinpoint the TSS but only indicates a region that might overlap or be in proximity with the promoter region.

You can use DPF's algorithm for promoter search in large contigs of anonymous DNA to make gene hunting easier because, in the search for general vertebrate promoters, it produces fewer FPs compared to other systems. ■

References

1. M. Scherf et al., "First Pass Annotation of Promoters on Human Chromosome 22," *Genome Research*, vol. 11, no. 3, Mar. 2001, pp. 333–340.
2. A.G. Pedersen et al., "The Biology of Eukaryotic Promoter Prediction: A Review," *Computers & Chemistry*, vol. 23, no. 3–4, June 1999, pp. 191–207.
3. M. Scherf, A. Klingenhoff, and T. Werner, "Highly Specific Localisation of Promoter Regions in Large Genomic Sequences by PromoterInspector: A Novel Context Analysis Approach," *J. Molecular Biology*, vol. 297, no. 3, Mar. 2000, pp. 599–606.
4. S. Knudsen, "Promoter2.0: For the Recognition of Pol II Promoter Sequences," *Bioinformatics*, vol. 15, no. 5, May 1999, pp. 356–361; www.cbs.dtu.dk/services/Promoter
5. M.G. Reese, N.L. Harris, and F.H. Eeckman, "Large-Scale Sequencing Specific Neural Networks for Promoter and Splice Site Recognition," *Proc. Pacific Symp.* (Biocomputing 96), World Scientific Publishing, Singapore, 1996; pp. 737–738; www.fruitfly.org/seq_tools/promoter.html.
6. R.C. Périer et al., "The Eukaryotic Promoter Database (EPD)," *Nucleic Acids Research*, vol. 28, no. 1, Jan. 2000, pp. 302–303.
7. D.A. Benson et al., "GenBank," *Nucleic Acids Research*, vol. 28, no. 1, Jan. 2000, pp. 15–18.
8. M. Reese et al., 1999; www.fruitfly.org/seq_tools/datasets/Human.
9. N. Mache, M. Reczko, and A. Hatzigeorgiou, "Multistate Time-Delay Neural Networks for the Recognition of Pol II Promoter Sequences," *Proc. 10th Conf. Intelligent Systems for Molecular Biology* (ISMB 96), St. Louis, 1996; www.informatik.uni-stuttgart.de/ipvr/bv/personen/mache/ismb/ismb.html.
10. J.W. Fickett and A.G. Hatzigeorgiou, "Eukaryotic Promoter Recognition," *Genome Research*, vol. 7, no. 9, Sept. 1997, pp. 861–878.