doi:10.1006/jmbi.2000.3589 available online at http://www.idealibrary.com on IDEAL® J. Mol. Biol. (2000) 297, 599–606

JMB

Highly Specific Localization of Promoter Regions in Large Genomic Sequences by PromoterInspector: A Novel Context Analysis Approach

Matthias Scherf[1], Andreas Klingenhoff[1,2] and Thomas Werner[1,2]*

[1]*Institute of Mammalian Genetics, GSF-National Research Center for Environment and Health Ingolstädter Landstraße 1 D-85758, Neuherberg Germany*

[2]*Genomatix Software GmbH Karlstrasse 55, D-80333 Munich, Germany*

**Corresponding author*

We present a new algorithm called PromoterInspector to locate eukaryotic polymase II promoter regions in large genomic sequences with a high degree of specificity. PromoterInspector focuses on the genetic context of promoters, rather than their exact location. Application of PromoterInspector can serve as a crucial pre-processing step for other methods to locate exactly, or to analyze promoters.

PromoterInspector does not depend on heuristics, because it is purely based on libraries of IUPAC words extracted from training sequences by an unsupervised learning approach.

We compared PromoterInspector to *in silico* promoter prediction tools using the sequences from the review by J. W. Fickett. PromoterInspector compared favourably on Fickett's evaluation scheme. A true positive to false positive ratio of 2.3 was obtained, surpassing the best ratio of 0.6, reported for TSSG. The application of our method to several large genomic sequences of over 1.3 million base-pairs in total resulted in even more specific predictions. The coverage of annotated promoters was comparable to other *in silico* promoter prediction methods, while the true positive predictions increased by up to 100 % of total matches.

PromoterInspector scans 100 kb in less than one minute on a workstation, and thus is especially applicable for large genome analysis. The method is available at
http://genomatix.gsf.de/cgi-bin/promoterinspector/promoterinspector.pl

Keywords: genome annotation; large-scale sequence analysis; genomic regulatory regions; *in silico* promoter prediction; unsupervised learning

Introduction

Within the next few years, the complete sequences of the human genome and several model organisms will put gigabases of new sequences into the public domain. Rapid and precise functional annotation of the collected large-scale sequences by *in silico* methods will be mandatory, since experimental analysis of potential genes and their regulation is clearly not possible on a full-genome scale. Therefore, it is essential to develop computer algorithms that can recognize genes and gene features, including promoter detection and characterization of eukaryotic promoter sequences.

Today, computational prediction of eukaryotic promoters solely from the nucleotide sequence is an attractive but difficult aspect of sequence analysis. Polymerase II promoters do not contain any sequence elements that are consistently shared. They usually consist of multiple binding sites for transcription factors that must occur in a specific context, apparently shared only by a small group of promoters (Werner, 1999). Thus, the combination and orientation of the transcription factors is the crucial information, rather than the mere occurrence of several binding sites. According to the individual architecture of polymerase II promoters, a general prediction strategy is not obvious. Consequently, a variety of different *in silico* promoter prediction tools exist.

These promoter prediction tools can roughly be divided in three groups: heuristic approaches; approaches that attempt to recognize core promoter elements such as TATA boxes, CAAT boxes

Abbreviations used: INR, transcription initiation sites; TSS, transcription start site.

E-mail address of the corresponding author: werner@gsf.de

0022-2836/00/030599–8 $35.00/0

and transcription initiation sites (INR); and approaches that attempt to use the whole ensemble of elements (transcription factor binding sites, oligonucleotides), found in a promoter.

Heuristic approaches use models that describe the orientation and context of several transcription factor binding sites and have been proven to be able to detect promoters with a very high level of specificity, but with limited coverage (Frech *et al*., 1997, 1998; Werner, 1999). Thus they are useful to predict specific promoter classes in large genomic sequences, but do not provide a general promoter prediction approach.

Approaches that attempt to recognize core promoter elements as well as methods that use frequencies of elements found in a promoter mostly predict of the order of one promoter per kilobase in human DNA (Fickett & Hatzigeorgiou, 1997). However, the average distance between functional promoters has been estimated to be in the range of 30 to 40 kb, with a very uneven distribution. Although some of the predicted promoters might correspond to cryptic initiation sites, it is likely that most of them are false positives. Some of the tools use a more restrictive approach to reduce the number of total predictions, but the problem of a huge number of false positive predictions remains (Fickett & Hatzigeorgiou, 1997). These large numbers of false positive matches are a problem precluding experimental verification. Thus, such methods will be of limited use for large-scale genomic sequences.

Results

PromoterInspector is motivated by the concept that polymerase II promoters are quite different in terms of individual organization, but are probably embedded into a common genomic context. Specific features of such a putative context are not yet known. Thus we base our prediction system on context features extracted from training sequences by an unsupervised learning technique.

Design of the prediction system

Definition of context features

Context features are based on an approach (Wolfertstetter *et al*., 1996) using oligonucleotides with one variable mismatch. We extended this approach by the introduction of wildcards at multiple positions. In more detail, context features are defined by disjunct groups of similar IUPAC words (IUPAC groups). Each IUPAC group is uniquely defined by a set of oligonucleotides and a number of undefined base-pairs (wildcards, ''N''). The IUPAC words of a IUPAC group contain all elements of the oligonucleotide set in the same order and orientation, and differ in the number of wildcards between them. The number of elements in a IUPAC group is defined by the number of possibilities of arranging the wildcards among the oligonucleotides. Wildcards at the beginning and end of IUPAC words are discarded. As an example, a IUPAC group which results from two wildcards and the oligonucleotide set (AGC, GCA) is (AGCGCA, AGCNGCA, AGCNNGCA).

Definition of decision instances

The prediction of the genomic promoter context is based on several decision instances (classifiers). A classifier gets a sequence of fixed length as input, and returns whether the sequence belongs to one of two classes.

A classifier is defined by two disjunct sets of IUPAC groups: a set of promoter-related IUPAC groups defines the class ''promoter'', while a set of non-promoter-related IUPAC groups defines the class ''non-promoter''.

The classification is based on IUPAC group matches. A IUPAC group matches in a sequence if at least one of its group members matches. The input sequence is assigned to the class ''promoter'' if the promoter-related IUPAC groups match more often in the sequence compared to the IUPAC groups of the other class.

The IUPAC group candidates of a classifier are directly extracted from a set of training sequences. All IUPAC groups that match at least once in these sequences are involved. More specifically, if IUPAC groups are defined by a set of two oligonucleotides of length two and two wildcards, the following candidates are formed from the training sequence AGCTG: (AGCT, AGNCT, AGNNCT), (AGTG, AGNTG, AGNNTG) and (GCTG, GCNTG, GCNNTG).

Promoter-specific and non-promoter-specific candidates are determined as follows: given a set of promoter and non-promoter sequences (training sequences), a candidate is assigned to the class promoter, if the ratio between the number of hits in the promoter and non-promoter training sequences exceeds a certain threshold and *vice versa*. Below we will refer to this threshold as the ''assignment threshold''.

Architecture of the prediction system

PromoterInspector is based on three classifiers. Each classifier is specialized to differentiate between promoter and one of the following non-promoter sequence sets: exon, intron and 3′-UTR. The prediction system assigns a sequence to the class promoter only if all three classifiers decide that the sequence belongs to this class.

Parameter optimization of the prediction system

According to the definitions above, the number of wildcards and the number and length of the elements in the oligonucleotide sets which define the IUPAC groups must be determined for every

classifier. Furthermore, an optimal assignment threshold must be calculated.

To reduce the large number of possible feature parameters, we define that the number of wildcards, as well as the number and length of the oligonucleotides, is identical for all IUPAC groups of a classifier. Parameters are optimized in four steps by brute force, through a threefold crossvalidation approach.

In the first step, the crossvalidation sets are prepared: from a given set of example sequences for every class, 90% of the sequences were used for parameter optimization (crossvalidation set), while 10% were kept for evaluation purposes (evaluation set). The crossvalidation set was split into three disjunct sets. From these sets, three different training sets were built by joining two of the sets in turn, while the third set was used as a test set.

In the second step, a set of different parameter constellations was generated. For every parameter, we defined an upper and lower limit, and chose parameters randomly within these intervals. The number of oligonucleotides was chosen between two and three, while the length of the oligonucleotides was set between three and five. The number of wildcards was set between one and six. Boundary limits for the assignment threshold were set between one and five.

In the third step, a classifier was built for every parameter constellation based on one of the three training sets, and the classifier which on average led to the best results on the crossvalidation test sets was kept. In the last step, the classifiers which resulted from step three were evaluated on the evaluation set. Table 1 shows the test and evaluation results.

The results of our experiments showed that the optimal assignment threshold for all classifiers is one. Thus, an IUPAC group is assigned to the class promoter whenever the number of hits of the IUPAC group in the promoter training sequences exceeds the number of hits in the training sequence of the class non-promoter, and *vice versa*. The correct prediction for promoter sequences decreased dramatically for larger assignment thresholds, while the correct predictions for non-promoter sequences increased slightly. This leads to the assumption, that a large assignment threshold leads to a small set of highly specific promoter features, which do not give a general description of the promoter context, but rather give a specific characterization of small promoter subgroups.

Application technique

Identification of promoter regions in large genomic sequences is performed by a sliding window approach. A window is moved over the sequence and its content is classified. A promoter region is reported if a certain number of consecutive windows are identified as members of the promoter class. Thus, three additional parameters, the length of the window, the offset between two consecutive windows and the number of consecutive hits, must be optimized.

We applied the same procedure as described above for the parameter optimization on the basis of crossvalidation and evaluation sets which contained sequences with more than 2000 bp respectively. The optimal window length was found to be 100, the optimal offset was found to be four and the optimal number of consecutive hits was 24.

The application of PromoterInspector led to the discovery of an interesting phenomenon: whenever a promoter region is predicted on the sense-strand, there is also a prediction of a promoter region on the antisense-strand at the same position. This result indicates that PromoterInspector indeed focuses on promoter regions rather than the orientation-specific promoters.

Application to Fickett's evaluation data set

The review data set used by Fickett & Hatzigeorgiou (1997) consists of 24 promoters covering a total of 33,120 bp. None of the sequences matched a sequence within the EPD (Cavin *et al*., 1998), neither at the level of identity nor at the level of clear homology (Fickett & Hatzigeorgiou, 1997).

Fickett evaluated Audic (Audic & Claverie, 1997), Autogene (Kondrakhin *et al*., 1994), GeneID (Knudson, 1999), NNPP 2.1 (Reese & Eeckmann, unpublished results), PromFind (Hutchinson, 1996), PromoterScan (Prestridge, 1995), TATA (Bucher, 1990), TSSG (Solovyev & Salamov, 1997) and TSSW (Solovyev & Salamov, 1997) with regard to their ability to approximately locate the transcription start site (TSS). In the case of tools that give a promoter region rather than predict the

Table 1. Results of the three PromoterInspector classifiers

	%TP predictions			
	Crossvalidation		Evaluation set	
Non-promoter set	Promoter	Non-promoter	Promoter	Non-promoter
Exon	82.6	70.6	80.3	75.1
Intron	57.6	80.8	60.3	81.2
3′-UTR	65.5	78.6	63.5	77.6

The three classifiers of PromoterInspector were applied to the crossvalidation test set and an evaluation set. The obtained true positive (TP) predictions are shown.

exact location of the TSS, he defined the 3′-end of a predicted region as the predicted TSS. A predicted TSS, explicit or implicit, was counted as correct if it was within 200 bp 5′, or 100 bp 3′ to an experimentally determined TSS.

PromoterInspector is not directly comparable to these other methods because the predicted promoter regions are not strand-specific. Thus, according to Fickett's evaluation criterion, PromoterInspector would predict a TSS at the 3′ as well as the 5′ end of a region, which would in most cases lead to at least one false positive TSS per predicted promoter region.

To make the comparison fairer, we based the evaluation of the methods on the assumption that the gene orientation is known, and treated all approaches as ''not strand specific'', i.e. all hits on the antisense strand with correct distance to the TSS according to the evaluation scheme of Fickett were also counted as true positives. PromoterInspector results were evaluated by Fickett's criterion and on an interval criterion. This interval criterion additionally defines a predicted region as true positive if the TSS lies within the region boundaries. The average length of the promoter regions predicted by PromoterInspector was 270 bp. Thus, only 3 % of the whole sequences were covered.

Results are shown in Table 2. For a detailed description of the methods we refer the reader to Fickett & Hatzigeorgiou (1997) and the original papers cited therein.

Analysis of large genomic sequences

PromoterInspector was applied to several large genomic sequences with more than 1.3 million bp in total. PromoterInspector predictions were compared to those of TSSG and TSSW (Solovyev & Salamov, 1997), NNPP 2.1 (Reese & Eeckmann, 1999) and Promoter 2.0 (Knudson, 1999). These methods were selected because they were accessible *via* the Internet, could handle large sequences, and exhibited acceptable computational speed. All methods were applied with default parameter values.

The results of the TSS prediction tools were analyzed on the basis of Fickett's evaluation scheme. Results of PromoterInspector were evaluated by the interval criterion mentioned above, i.e. a region was counted as true positive if a TSS is located within the region, or a region boundary is at most 200 bp 5′ to a TSS. An overview of the sequences, their length and the number of annotated TSS in the sequence is shown in Table 3 while the results are presented in Table 4. Figure 1 is a graphical representation of the summary shown in Table 5.

The regions predicted by PromoterInspector covered 1.1 % of the long genomic sequences. The length of the regions was, on average, 438 bp and ranged from 198 bp to 1348 bp.

Discussion

The purpose of PromoterInspector is to identify regions containing promoters based on a common genomic context of polymerase II promoters. The results of PromoterInspector underline the practicability of our approach in the analysis of large genomic sequences. The experiments demonstrated that 43 % of PromoterInspector predictions can be expected to be true positives, while 43 % of the annotated TSS were predicted correctly. Therefore, it is presented as an important pre-processing step for other methods focusing on TSS prediction. However, these are pessimistic results, since the number of genes in the sequences exceeds the number of annotated TSS, which is, for example, true for the sequence described by Chen *et al*. (1996).

PromoterInspector yielded good results with Fickett's evaluation scheme, although it is not designed for exact promoter prediction. Thus, comparison of PromoterInspector and the TSS prediction methods primarily illustrates the limits of

Table 2. Results for the Fickett data set

Method	TP	%TP of total matches	FP	TP/FP	Coverage (%)
Audic (Audic & Claverie, 1997)	9	24	29	0.31	37
Autogene (Kondrakhin *et al*., 1994)	8	15	46	0.17	33
GeneID (Knudson, 1999)	11	19	48	0.23	46
NNPP 2.1 (Reese & Eeckmann, unpublished results)	14	19	59	0.24	58
PromFind (Hutchinson, 1996)	11	31	24	0.46	46
PromoterScan (Prestridge, 1995)	3	33	6	0.50	12
TATA (Bucher, 1990)	7	23	24	0.30	29
TSSG (Solovyev & Salamov, 1997)	10	37	17	0.60	42
TSSW (Solovyev & Salamov, 1997)	14	30	33	0.42	58
PromoterInspector (Fickett evaluation)	7	70	3	2.3	29
PromoterInspector (Intervall evaluation)	9	90	1	9.0	37

Number of true positive (TP) and false positive (FP) predictions were calculated based on the assumption, that the gene orientation is known, i.e. if the experimental TSS is found on the sense strand of the sequence, only TSS predictions on the sense strand were considered. Coverage shows the percentage of all experimentally verified TSS which were correctly predicted by the methods.

Table 3. Description of the large genomic sequences which were included in the analysis

Accession number	Description	Length (bp)	Number of TSS
AC002397	Complete sequence of mouse chromosome 6 BAC-284H12 (Ansari-Larei *et al.*, 1998)	227,538	17
L44140	*Homo sapiens* chromosome X region from filamin (FLN) gene to glucose-6-phosphate dehydrogenase (G6PD) gene. There are 13 known and six candidate genes in the sequence (Chen *et al.*, 1996)	219,447	11
D87675	*Homo sapiens* DNA for amyloid precursor protein	301,692	1
AF017257	*Homo sapiens* chromosome 21-derived BAC containing erythroblastosis virus oncogene homolog 2 protein (ets-2) gene	101,569	1
AF146793	Mouse protein B, Clock, PFT27 and HSAR gene	204,625	4
AC002368	*Homo sapiens* Xq28 BAC PAC and cosmid clones containing FMR2 gene	324,816	1
Total		1.37 million	35

specificity with general methods for exact promoter location, the reason why we concentrated on context features to predict promoter regions.

However, specific features of the genomic promoter context are so far unknown. As a consequence, an unsupervised strategy was used to build PromoterInspector as a heuristc-free prediction system. Such a technique has been applied in several methods of *in silico* promoter prediction and analysis but with different objective targets.

One of the most general heuristic-free methods is Xlandscape (Levy *et al.*, 1998). Xlandscape looks up every oligonucleotide (word) of a given sequence in a set of databases and displays the resulting word frequencies with respect to their occurrence in the sequence. Xlandscape addresses a wide variety of tasks such as the identification of repeat regions, similarity check between sequences, preference-check of words from different gene sections such as exons or introns, as well as the identification of word overrepresentations. Levy *et al.*

Table 4. Results of the large genomic sequence analysis

Sequence accession number	Method	TP	FP	%TP total predctions	% Coverage
AC002397	TSSG	8	97	7.6	47
	TSSW	6	101	5.6	35
	NNPP 2.1	10	658	1.5	58
	Promoter 2.0	5	455	1.1	29
	PromoterInspector	5	1	83.3	29
L44140	TSSG	4	175	2.2	36
	TSSW	4	184	2.1	36
	NNPP 2.1	8	554	1.4	73
	Promoter 2.0	2	353	0.5	18
	PromoterInspector	6	14	30.0	55
D87675	TSSG	0	36	0	0
	TSSW	1	36	2.7	100
	NNPP 2.1	1	427	0.2	100
	Promoter 2.0	0	222	0	0
	PromoterInspector	1	2	50.0	100
AF017257	TSSG	0	16	0	0
	TSSW	1	20	5.0	100
	NNPP 2.1	0	141	0	0
	Promoter 2.0	0	85	0	0
	PromoterInspector	1	0	100.0	100
AF146793	TSSG	2	59	3.2	50
	TSSW	2	75	2.6	50
	NNPP 2.1	3	674	0.4	75
	Promoter 2.0	1	273	0.3	25
	PromoterInspector	2	1	66.7	50
AC002368	TSSG	1	66	1.5	100
	TSSW	1	85	1.3	100
	NNPP 2.1	1	1079	0.1	100
	Promoter 2.0	0	373	0	0
	PromoterInspector	1	1	50.0	100

Evaluation of the TSS predictions of TSSG, TSSW, NNPP 2.1 and Promoter 2.0 was according to the evaluation scheme by Fickett. A region predicted by PromoterInspector was counted as correct if a TSS was located within the region or if a region boundary was within 200 bp 5′ of such a TSS.

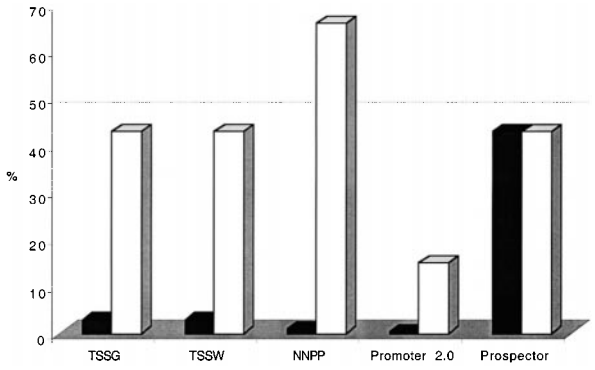

Figure 1. Summary of large-scale sequence analysis results. The quality of prediction by the five methods evaluated is represented by two values. The percentage of true positive matches to the overall number of predictions (black column) and the percentage of true positive predictions to the overall number of annotated transcription starts (coverage, white column). Values are based on predictions and annotations of the six large genomic sequences. (Prospector = PromoterInspector.)

(1998) applied Xlandscape to the eukaryotic promoter database version 48 and showed a correlation between overrepresented words and potential transcription factor binding sites.

PromFD (Chen *et al.*, 1997) uses words of different lengths as well as information matrices to predict promoters. Word selection takes advantage of the fact that transcription factor binding sites are frequently detected in clusters. Furthermore, the overall frequency of words in promoter and non-promoter sequences is taken into consideration.

PromFind (Hutchinson, 1996) predicts promoter regions on the basis of words of length six. Words are selected by their different frequencies between promoter to non-coding, as well as promoter to coding, sequences.

All the methods mentioned above have a completely different focus from PromoterInspector. Xlandscape focuses on the visualization of word frequency analyses rather than prediction strategies. The aim of PromFD is to predict the exact location of a promoter, while PromFind shares the goal of predicting promoter regions but can do so only under several assumptions. To quote Hutchinson (1996): ''The program is not designed to determine whether a promoter region exists within an unknown sequence, but rather assumes that there is a promoter and identifies the most probable region containing it''.

A comparison between PromoterInspector and the methods PromFD and PromFind elucidates two differences: (i) the predictions of the two methods are based on words rather than IUPAC groups; and (ii) both methods use only one set of non-promoter sequences, and therefore one classifier, rather than subdividing the set into subgroups and using several classifiers.

The probability of occurrence of a word decreases exponentially with the length of the word (assuming equal distribution of nucleotides). Thus it is difficult to get a good balance of sensitivity and specificity when the decision is purely based on the occurrence of single words.

In the PromoterInspector approach, this problem is addressed by IUPAC groups, which represent a word by a set of at least two similar IUPAC words. Consequently, a IUPAC group match is more likely than a match of the word represented by the IUPAC group. The effect of the IUPAC group strategy is shown in Figure 2. A classifier for the discrimination of promoter and exon sequences was trained with different numbers of IUPAC group elements. These classifiers were applied to an evaluation set with promoter and exon sequences. The number of IUPAC group members was plotted against the observed specificity and sensitivity for promoter prediction. IUPAC groups with one element contain a word without wildcards, and thus correspond to approaches such as Xlandscape, PromFD or PromFind. The Figure shows an improved performance of IUPAC groups with two IUPAC words compared to those with only one member. Furthermore, it is obvious that the introduction of IUPAC groups addresses the problem of balancing the sensitivity and specificity of predictions. In this example, a 25 % gain in specificity was obtained while the loss in sensitivity was very low (10 %). However, the IUPAC group approach is based on the assumption, that a small variation in a word does not significantly change its information content, although this might not necessarily be true for all applications.

Table 5. Summary of large genomic sequence analysis

Method	TP	FP	% TP of total matches	% Coverage
TSSG	15	449	3.2	43
TSSW	15	501	2.9	43
NNPP 2.1	23	3533	0.6	66
Promoter 2.0	8	1751	0.4	15
PromoterInspector	15	19	43	43

Values represent the grand total of all true positive and false positive predictions which were obtained by applying the methods to the six large genomic sequences (1.37 million bp).

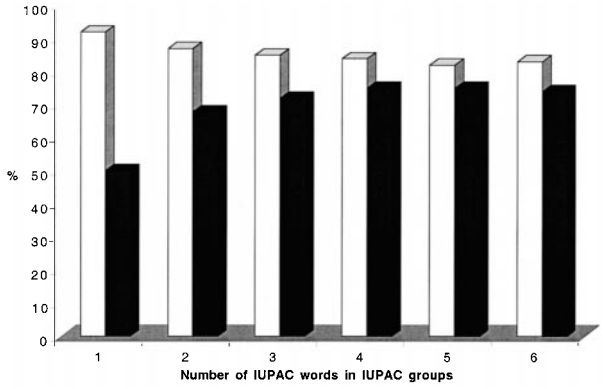

Figure 2. Influence of number of IUPAC group elements on sensitivity and specificity of promoter prediction. Results are obtained from a classifier which discriminates exon and promoter sequences. The classifier design was based on different numbers of IUPAC group elements and applied to a promoter and an exon evaluation set. Black columns represent the specificity and white columns the sensitivity of the classifier according to the promoter prediction in the evaluation set. An IUPAC group with one element contains an oligonucleotide and is therefore comparable to methods such as PromFD or PromFind.

PromoterInspector is based on three classifiers, which are specialized to differentiate between promoter regions and a subset of non-promoter sequences (intron, exon and 3′-UTR). In contrast to this, PromFD and PromFind use only one classifier, i.e. the features are extracted from one promoter set and one set of various non-promoter sequences. To compare these two approaches, we built two versions of PromoterInspector. Version V1 was based on one set of mixed non-promoter sequences, while version V2 was built on the basis of exon, intron and 3′-UTR as described above. In both versions, all IUPAC groups were based on words of equal length. Both versions of PromoterInspector were applied to exon, intron, 3′-UTR and promoter evaluation sequence sets. As a result, we obtained a false positives ratio between V1 and V2 of 1.4, while the ratio of true positive promoter predictions was 0.3. Thus, V1 is less sensitive and less specific. Furthermore the three classifiers of V2 differ significantly according to their sets of promoter and non-promoter IUPAC groups.

PromoterInspector allows for several new developments and research opportunities for *in silico* promoter prediction. We will next test combinations of the PromoterInspector approach with TSS prediction methods to improve their specificity. Besides this, a combination with tools for the detection of promoter-specific features such as transcription factor binding sites or promoter models might also lead to improved results. Furthermore, we will apply data mining tools to analyze the extracted features and compare the three classifiers of PromoterInspector.

Materials and Methods

Sequence training sets

From the vertebrate section of the eukaryotic promoter database (EPD), V 60.0. (Cavin *et al*., 1998), promoter sequences from 500 bp upstream to 50 bp downstream of the TSS were taken. Vertebrate exon and vertebrate intron sequences of different location, covering a total of 1 million bp in each set were randomly extracted from GenBank contents in August 1999. Vertebrate 3′-UTR sequences with a total of 1 million bp were extracted from the UTR database (Pesole *et al*., 1999).

Data preprocessing

Each of the three classifiers handles sequences of 100 bp length. Example sets for the four sequence classes (promoter, intron, exon and 3′-UTR) were created by randomly extracting non-overlapping sequences of 100 bp from the four example sets mentioned above. Redundant sequences were deleted by the program CLEANUP (Grillo *et al*., 1996) which resulted in sets consisting of 2107 sequences from promoter regions, 6787 exon sequences, 8570 intron sequences and 8562 3′-UTR sequences.

Selection of large genomic sequences

Large genomic sequences were selected from the EMBL database (Stoesser *et al*., 1999). The annotated genes in sequences AC002368, AC002397 and AF146793 were identified by similarity search. Annotations for sequences AF017257, D87675 and L44140 are experimentally verified. A total of six of the 35 genes in the sequences are annotated to be 5′ incomplete. None of the genes is annotated to be based on *in silico* prediction.

Acknowledgments

We thank Ralf Schneider, Korbinian Grote, Kerstin Quandt, Kornelie Frech and Valérie Gailus-Durner for fruitful discussions and careful reading of the manuscript. This work was supported by the BMBF Project FANGREB 51440030311641.

References

Ansari-Lari, M. A., Oeltjen, J. C., Schwartz, S., Zhang, Z., Muzny, D. M., Lu, J., Gorrell, J. H., Chinault, A. C., Belmont, J. W., Miller, W. & Gibbs, R. A. (1998). Comparative sequence analysis of a gene-rich cluster at human chromosome 12p13 and its syntenic region in mouse chromosome 6. *Genome Res.* **8**, 29-40.

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

Bucher, P. (1990). Weight matrix descriptions of four eukaryotic RNA polymerase II promoter elements derived from 502 unrelated promoter sequences. *J. Mol. Biol.* **212**, 564-575.

Cavin, Périer R., Junier, T. & Bucher, P. (1998). The Eukaryotic Promoter Database EPD. *Nucl. Acids Res.* **26**, 353-357.

Chen, E. Y., Zollo, M., Mazzarella, R. A., Ciccodicola, A., Chen, C.-N., Zuo, L., Heiner, C., Burough, F. W., Ripetto, M., Schlessinger, D. & D'Urso, M. (1996). Long-range sequence analysis in Xq28: thirteen known and six candidate genes in 219.4 kb of high GC DNA between the RCP/GCP and G6PD loci. *Hum. Mol. Genet.* **5**, 659-668.

Chen, Q., Hertz, G. & Stormo, G. D. (1997). PromFD 1.0: a computer program that predicts eukaryotic pol II promoters using strings and IMD matrices. *Comput. Applic. Biosci.* **13**, 29-35.

Fickett, J. W. & Hatzigeorgiou, A. G. (1997). Eukaryotic promoter recognition. *Genome Res.* **7**, 861-878.

Frech, K., Danescu-Mayer, J. & Werner, T. (1997). A novel method to develop hghly specific models for regulatory units detects a new LTR in GenBank which contains a functional promoter. *J. Mol. Biol.* **270**, 674-687.

Frech, K., Quandt, K. & Werner, T. (1998). Muscle actin genes: a first step towards computational classification of tissue specific promoters. *In Silico Biol.* **1**, 29-38.

Grillo, G., Attimonelli, M., Liuni, S. & Pesole, G. (1996). CLEANUP: a fast computer program for removing redundancies from nucleotide sequence databases. *Comput. Applic. Biosci.* **12**, 1-8.

Hutchinson, G. (1996). The prediction of vertebrate promoter regions using differential hexamer frequency analysis. *Comput. Applic. Biosci.* **12**, 391-398.

Knudsen, S. (1999). Promoter 2.0: for the recognition of Pol II promoter sequences. *Bioinformatics,* **15**, 356-361.

Kondrakhin, Y., Shamir, Y. & Kolchanov, N. (1994). Construction of a generalized consensus matrix for recognition of vertebrate pre-mRNA 3′ terminal processing sites. *Comput. Applic. Biosci.* **10**, 597-603.

Levy, S., Compagnoni, L., Myers, E. W. & Stormo, G. D. (1998). Xlandscape: the graphical display of word frequencies in sequences. *Bioinformatics,* **14**, 74-80.

Pesole, G., Liuni, S., Grillo, G., Ippedico, M., Larizza, A., Makalowski, W. & Saccone, C. (1999). UTRdb: a specialized database of 5′ and 3′ untranslated regions of eukaryotic mRNAs. *Nucl. Acids Res.* **27**, 188-191.

Prestridge, D. (1995). Predicting pol II promoter sequences using transcription factor binding sites. *J. Mol. Biol.* **249**, 923-932.

Solovyev, V. & Salamov, A. (1997). The gene-finder computer tools for analysis of human and model organisms genome sequences. *ISMB97,* 294-302.

Stoesser, G., Tuli, M. A., Lopez, R. & Sterk, P. (1999). EMBL Outstation - The European Informatics Institute. *Nucl. Acids Res.* **27**, 18-24.

Werner, T. (1999). Models for prediction and recognition of eukaryotic promoters. *Mammalian Genome,* **10**, 168-175.

Wolfertstetter, F., Frech, K., Herrmann, G. & Werner, T. (1996). Identification of functional elements in unaligned nucleic acid sequences by a novel tupel search algorithm. *Comput. Applic. Biosci.* **12**, 71-80.

Edited by J. Karn

(Received 10 December 1999; received in revised form 3 February 2000; accepted 9 February 2000)