JOURNAL OF COMPUTATIONAL BIOLOGY
Volume 13, Number 2, 2006

Pp. 379–393

Using Multiple Alignments to Improve Gene Prediction

SAMUEL S. GROSS and MICHAEL R. BRENT

ABSTRACT

The multiple species *de novo* gene prediction problem can be stated as follows: given an alignment of genomic sequences from two or more organisms, predict the location and structure of all protein-coding genes in one or more of the sequences. Here, we present a new system, N-SCAN (a.k.a. TWINSCAN 3.0), for addressing this problem. N-SCAN can model the phylogenetic relationships between the aligned genome sequences, context-dependent substitution rates, and insertions and deletions. An implementation of N-SCAN was created and used to generate predictions for the entire human genome and the genome of the fruit fly *Drosophila melanogaster*. Analyses of the predictions reveal that N-SCAN's accuracy in both human and fly exceeds that of all previously published whole-genome *de novo* gene predictors.

Key words: gene prediction, genome annotation, comparative genomics, phylogenetic models.

1. INTRODUCTION

TWO RECENT DEVELOPMENTS HAVE INCREASED INTEREST in *de novo* gene prediction. First, the availability of assemblies of several nonhuman vertebrate genomes has created the possibility for further significant improvements in human gene prediction through the use of comparative genomics. Second, traditional experimental methods for identifying genes based on 5′ EST sampling and cDNA clone sequencing are now reaching the point of diminishing returns far short of the full gene set (Gerhard *et al.*, 2004). As a result, efforts to identify new genes by RT-PCR from predicted gene structures are taking on greater importance.

A major advantage of *de novo* gene predictors is that they do not require cDNA or EST evidence or similarity to known transcripts when making predictions. This allows them to predict novel genes not clearly homologous to any previously known gene, as well as genes that are expressed at very low levels or in only a few specific tissue types, which are unlikely to be found by random sequencing of cDNA libraries. *De novo* gene predictors are therefore well suited to the task of identifying new targets for RT-PCR experiments aimed at expanding the set of known genes.

One of the first *de novo* systems to perform well on typical genomic sequences containing multiple genes in both orientations was GENSCAN (Burge and Karlin, 1997). GENSCAN uses a generalized hidden Markov model (GHMM) to predict genes in a given target sequence, using only that sequence as input. GENSCAN remained one of the most accurate and widely used systems prior to the advent of

Department of Computer Science and Engineering, Washington University, St. Louis, MO 63130.

dual-genome *de novo* gene predictors. The initial sequencing of the mouse genome made it possible for the first time to incorporate whole-genome comparison into human gene prediction (Waterston *et al.*, 2002). This led to the creation of a new generation of gene predictors, such as SLAM (Alexandersson *et al.*, 2003), SGP2 (Parra *et al.*, 2003), and TWINSCAN (Korf *et al.*, 2001; Flicek *et al.*, 2003; Tenney *et al.*, 2004) which were able to improve on the performance of GENSCAN by using patterns of conservation between the human and mouse genomes to help identify coding regions, splice sites, and other signal sequences with functions in transcription and translation. Until now, these programs have been the most accurate *de novo* gene predictors for mammalian genomes.

Recently, there has been an effort to create systems capable of using information from several aligned genomes to further increase predictive accuracy beyond what is possible with two-genome alignments. Programs such as EXONIPHY (Siepel and Haussler, 2004a), SHADOWER (McAuliffe *et al.*, 2003), and those based on EHMMs (Pedersen and Hein, 2003) fall into this category. While many important advances have been made in this area, no system of this type has yet managed to robustly outperform two-sequence systems on a genomic scale.

The gene prediction system presented here, N-SCAN (or TWINSCAN 3.0), extends the TWINSCAN model to allow for an arbitrary number of informant sequences as well as richer models of sequence evolution. N-SCAN is descended from TWINSCAN 2.0, which is in turn descended from the GENSCAN GHMM framework. However, instead of emitting a single DNA sequence like GENSCAN or a target DNA sequence and a conservation sequence like TWINSCAN, each state in the N-SCAN GHMM emits columns of a multiple alignment. N-SCAN uses output distributions for the target sequence that are similar to those used by TWINSCAN 2.0 and GENSCAN. It augments these with Bayesian networks which capture the evolutionary relationships among organisms in the multiple alignment. The model is also augmented with states for noncoding exons in the 5′ UTR and a state for conserved intergenic sequences.

2. METHODS

2.1. Overview

Whereas the GHMM underlying TWINSCAN 2.0 outputs a target genomic sequence and a conservation sequence, N-SCAN's GHMM outputs a multiple alignment whose rows consist of a target sequence and N informant sequences. The target sequence is made up of the four DNA bases, while the informant sequences can also contain the character "_" representing gaps in the alignment and "." representing positions in the target sequence to which the informant sequence does not align. The states in the N-SCAN GHMM correspond to functional categories in the target sequence only. Therefore, N-SCAN annotates only one sequence at a time. If annotations are desired for more than one of the sequences in the alignment, the system can be run multiple times with different sequences designated as the target.

One component of the model defines, for each GHMM state, the probability

$$P(T_i|T_{i-1}\ldots T_{i-d}) \tag{1}$$

of outputting a particular base in the target genome, given the previous d bases. Here, $T_1 \ldots T_L$ is the full target sequence $\mathbf{T}$, and d is the model order. This probability is implicitly dependent on the GHMM state at base i. States can be thought of as representing functional features of the sequence, such as start and stop codons, splice sites, and coding sequence. N-SCAN uses these target genome models in combination with a set of Bayesian networks to define, for each state, the probability

$$P(T_i, \mathbf{I_i}|T_{i-1}\ldots T_{i-d}, \mathbf{I_{i-1}}\ldots \mathbf{I_{i-o}}) \tag{2}$$

of outputting a column in the alignment given the previous d positions in the target and the previous o positions in the informants. Here $\mathbf{I_i} = \{I_i^1 \ldots I_i^N\}$ is the set of all informant characters at position i, and o is the informant model order. Expression (2) is calculated by multiplying the probability from the target genome model by

$$P(\mathbf{I_i}|T_i \ldots T_{i-d}, \mathbf{I_{i-1}}\ldots \mathbf{I_{i-o}}). \tag{3}$$

This quantity can be computed from the Bayesian network associated with the state.

We assume that the probability of outputting a base in the target sequence is independent of the values of the previous o positions in all of the informants, given the values of the previous d positions in the target. That is,

$$P(T_i|T_{i-1}\ldots T_{i-d}) = P(T_i|T_{i-1}\ldots T_{i-d}, \mathbf{I_{i-1}}\ldots\mathbf{I_{i-o}}). \quad (4)$$

Given (4), we can multiply (1) by (3) to obtain (2).

2.2. Phylogenetic Bayesian networks

The Bayesian network representation used in N-SCAN is similar to the phylogenetic models described by Siepel and Haussler (2004b), with a few important differences. First, the N-SCAN model uses a six-character alphabet consisting of the four DNA bases plus characters representing gaps and unaligned positions. Second, the substitutions between nodes in the model need not take place via a continuous time Markov process, nor must the substitution process be homogenous throughout the tree. Finally, the N-SCAN models make use of a different factorization of the joint distribution over model variables. For now, we focus on zeroth order substition models in which each column is independent of the others, given the state from which it was emitted. Higher order models will be introduced later.

Consider a phylogenetic tree such as the one shown in Fig. 1, left. Leaf nodes represent present-day species, while nonleaf nodes represent ancestral species which no longer exist. The same graph can also be interpreted as a Bayesian network describing a probability distribution over columns in a multiple alignment. In that case, the nodes represent random variables corresponding to characters at specific rows in a multiple alignment column, and the edges encode conditional independence relations among the variables. The independencies represented by the phylogenetic tree are quite natural—once we know the value of the ancestral character at a particular site in the alignment column, the probabilities of the characters in one descendant lineage are independent of the characters in other descendant lineages. These independence relations allow us to factor the joint distribution as follows:

$$P(H, C, M, R, A_1, A_2, A_3) = P(A_1)P(C|A_1)P(A_2|A_1)P(H|A_2)P(A_3|A_2)P(M|A_3)P(R|A_3).$$

By taking advantage of the conditional independence relations present in the seven-variable joint distribution, we can express it as a product of six local conditional probability distributions (CPDs) that have two variables each along with one marginal distribution. In general, factoring according to the independencies represented by a phylogenetic tree leads to an exponential reduction in the number of parameters required to specify the joint distribution. Of course, a real multiple alignment will consist only of sequences from currently existing species. Therefore, we treat the ancestral variables as missing data for the purposes of training and inference (see below). Rather than using a Bayesian network with the same structure as the phylogenetic tree, however, we apply a transformation to the phylogenetic tree graph to create the Bayesian network structure used in N-SCAN.

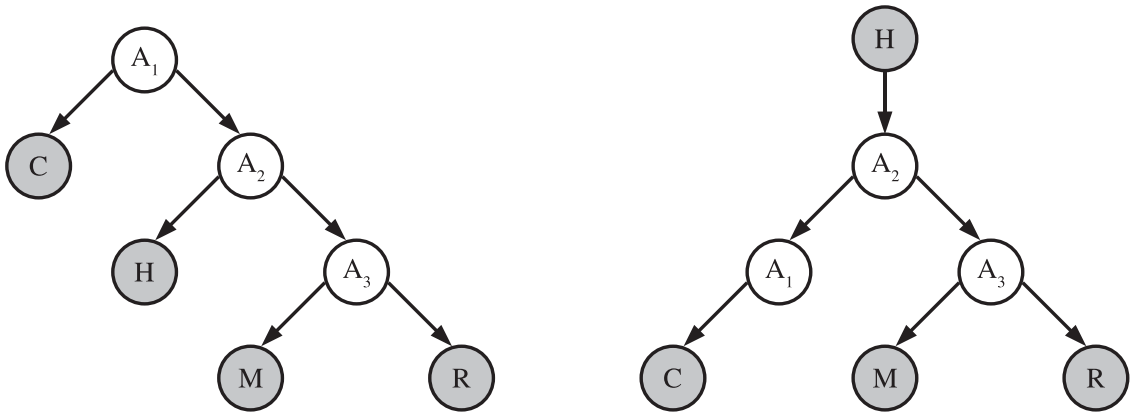

FIG. 1. A phylogenetic tree relating chicken (C), human (H), mouse (M), and rat (R). The graph can also be interpreted as a Bayesian network (**left**). The result of transforming the Bayesian network (**right**).

To transform the graph, we reverse the direction of all the edges along the path from the root of the graph to the target node. This results in a new Bayesian network with a tree structure rooted at the target node (Fig. 1, right). The new Bayesian network encodes the same conditional independence relations as the original, but it suggests a new factorization of the joint distribution. For the example network shown in Fig. 1, this factorization is

$$P(H, C, M, R, A_1, A_2, A_3) = P(H)P(A_2|H)P(A_1|A_2)P(A_3|A_2)P(C|A_1)P(M|A_3)P(R|A_3).$$

In this factorization, the local distribution at the node corresponding to the target sequence ($P(H)$ in the example) is not conditioned on any of the other variables. This allows us to directly use existing single-sequence gene models to account for the effect of the target sequence on the likelihoods of the functional states. Previous attempts to integrate phylogenetic trees and HMMs have used a prior distribution on the unobserved common ancestor sequence.

One final alteration to the Bayesian network is made after the transformation described above. Any ancestral node with just one child is removed from the network along with its impinging edges. For each removed node, a new edge is added from the removed node's parent to its child. In the example, we remove A_1 and add an edge from A_2 to C. Although the new network has one fewer variable, it can still represent any marginal distribution over the observed variables that the original network could represent. This follows from the fact that we can write the local CPD at the removed node's child as a sum over all the possible values of the removed node. In the example,

$$P(C|A_2) = \sum_{A_1} P(C|A_1)P(A_1|A_2).$$

In effect, we have implicitly summed out some of the unobserved variables in the distribution. In general, we are interested only in computing the probability of an assignment to the observed variables. When making such a computation, we explicitly sum out all the unobserved variables in the distribution. The transformation described above makes this computation more efficient by reducing the number of explicit summations required.

2.3. Higher order models

Following Siepel and Haussler (2004b), we can extend the models presented above to incorporate context dependence by redefining the meaning of the variables in the network. For a model of order o, we interpret the random variables in the network to represent the value of $o + 1$ adjacent positions in a row in the alignment. The entire network then defines a joint distribution over sets of $o + 1$ adjacent columns, which can be used to determine the probability of a single column given the previous o columns.

Inference in the network can be accomplished using a modified version of Felsenstein's algorithm (Felsenstein, 1982). For a given assignment, we define $L_u(a)$ to be the joint probability of all the observed variables that descend from node u, given that node u has value a. If $C(u)$ is the set of children of u and $V(u)$ is the set of possible values of u, we can calculate $L_u(a)$ according to the following recursive formula:

$$L_u(a) = \begin{cases} M(u, a) & \text{if } u \text{ is a leaf} \\ \displaystyle\prod_{c \in C(u)} \left(\sum_{b \in V(c)} Pr(c = b|u = a) L_c(b) \right) & \text{otherwise.} \end{cases}$$

Here, M is called the match function, and is defined as follows:

$$M(u, a) = \begin{cases} 1 & \text{if node } u \text{ has value } a \\ 0 & \text{otherwise.} \end{cases}$$

If T is the target node and t is its observed assignment, then $L_T(t)$ is the probability of the informant assignments given the target. To calculate all the L_u's for each node in the network, we can visit the nodes in postorder and calculate all the L_u's for a particular node at once.

We can use essentially the same algorithm for a conditional probability query. We define the partial match function, M', for a model of order o as follows:

$$M'(u, a) = \begin{cases} 1 & \text{if the first } o \text{ characters of the value of node } u \text{ match the first } o \text{ characters of } a \\ 0 & \text{otherwise.} \end{cases}$$

We define the quantity $L'_u(a)$ exactly as we did $L_u(a)$, except we substitute M' for M in the recursive definition. $L'_T(t)$ is then the probability of the first o characters of the informant assignments. Once we know the values of $L_T(t)$ and $L'_T(t)$, expression (3) is just

$$\frac{L_T(t)}{L'_T(t)}.$$

Each call to the inference algorithm visits each node in the network exactly once and requires $O(6^{2(o+1)})$ operations per internal node. Thus, the overall time complexity of inference is $O(N \cdot 6^{2(o+1)})$. Adding additional informants results only in a linear increase in the complexity of inference, but we pay an exponential cost for increasing the model order.

2.4. Training

The Bayesian networks for all of N-SCAN's GHMM states share a single topology determined by the phylogenetic tree relating the target and the informant genomes, which we assume to be known. However, the local CPDs for each node in a particular network will depend on the GHMM state with which the network is associated. The CPDs are not known in advance and must be estimated from training data.

Suppose we had a multiple alignment of all the genomes represented in the phylogenetic tree, with each column labeled to indicate which GHMM state produced it. For a particular Bayesian network of order o, we could treat each set of $o + 1$ adjacent columns ending with a column labeled by the GHMM state associated with the network as an instantiation of the network variables.

Once we extract a list of all the instantiations that occur in the multiple alignment, along with the number of times each instantiation occurs, it is a simple matter to produce a maximum likelihood estimate for all the CPDs in the network.

Since the GHMM states correspond to gene features, we can construct a labeled multiple alignment by combining the output of a whole-genome multiple aligner with a set of annotations of known genes in the target genome. However, the alignment will contain only the genomes that correspond to the root and leaves of the Bayesian network graph. The ancestral genomes are no longer available for sequencing and so must be treated as missing data.

We can still estimate the CPDs despite the missing data by using the EM algorithm. For each network, we begin with an initial guess for the CPDs. We then calculate, for each CPD, the expected number of times each possible assignment to its variables occurs in the multiple alignment. This can be done efficiently using a variation of the inside–outside algorithm presented by Siepel and Haussler (2004b), modified to take into account the fact that the root node in the N-SCAN Bayesian networks has just one child and that its value is observed. Next, the initial guess is replaced with a maximum likelihood estimate of the CPDs based on the expected occurrences. This process is repeated until the maximum likelihood estimate converges on a local maximum.

2.5. CPD parameterizations

We have not yet described a method for obtaining a maximum likelihood estimate of the CPDs from a set of observations (or expected observations). If no restrictions are placed on the form taken by the CPDs, there exist simple closed-form expressions for the value of each entry in each CPD. However, a network of order o with unrestricted CPDs has $(2N-1)(6^{o+1})(6^{o+1}-1)$ free parameters. If the amount of training data (i.e., columns in the multiple alignment with the appropriate label) available is small, this may be too many parameters to fit accurately.

The number of parameters can be reduced by specifying the CPDs with fewer than $(6^{o+1})(6^{o+1}-1)$ parameters each. Only a subset of all possible CPDs will be expressible by any given nongeneral parameterization, but we hope the maximum likelihood CPDs, or ones close to them, will be expressible

by the parameterization we choose. Depending on the parameterization chosen, we may be able to derive analytical expressions for the maximum likelihood estimates of the parameters. Otherwise, numerical optimization techniques must be used.

In the experiments below, we use a parameterization whose form is similar to that of the general reversible rate matrices used in traditional phylogenetic models. The zero-order version of this parameterization, which we call a *partially reversible* model, is shown below. A cell (i, j) in the matrix represents $P(j|i)$, the probability of a particular child node having value j from the alphabet {A, C, G, T, _, .}, given that its parent has value i.

$$\begin{pmatrix} - & a\pi_C & b\pi_G & c\pi_T & g & h \\ a\pi_A & - & d\pi_G & e\pi_T & g & h \\ b\pi_A & d\pi_C & - & f\pi_T & g & h \\ c\pi_A & e\pi_C & f\pi_G & - & g & h \\ i\pi_A & i\pi_C & i\pi_G & i\pi_T & - & j \\ k\pi_A & k\pi_C & k\pi_G & k\pi_T & l & - \end{pmatrix}$$

Here, the π_i's are the background frequency of the bases; they are estimated directly from the multiple alignment and are not considered to be free parameters. The model has 12 free parameters, as opposed to 30 in a general parameterization. The 4×4 upper-left submatrix is identical to the general reversible rate matrix used in continuous time Markov process models of sequence evolution (Lió and Goldman, 1998). The probability of a deletion is the same for each base, as is the probability of a base becoming unaligned. The probability of a base being inserted or becoming realigned is proportional to the background frequency of the base.

To generalize the partially reversible parameterization to higher orders, we make use of the concept of a gap pattern. We define the gap pattern of an $(o + 1)$-mer to be the string that results from replacing all the bases in the $(o + 1)$-mer with the character X. For example, the trimers GA_, GC_, and AT_ all have the gap pattern XX_. For substitution probabilities involving an $(o+1)$-mer that contain gaps or unaligned characters, the partially reversible model considers only the gap pattern of the $(o + 1)$-mer. Let $\mathcal{D}$ be the set of all possible $(o + 1)$-mers that contain only the four DNA bases, and $\mathcal{G}$ be the set of all possible $(o+1)$-mers that contain at least one gap or unaligned character. Let $G(i)$ be the gap pattern of $(o+1)$-mer i. The substitution probabilities $P(j|i)$ have the following properties:

1. If $j \in \mathcal{D}$ and $i \in \mathcal{D}$, then $P(j|i)\pi_i = P(i|j)\pi_j$.
2. If $j \in \mathcal{D}$ and $i \in \mathcal{G}$, then $P(j|i) = \alpha_{G(i)}\pi_j$.
3. If $j \in \mathcal{G}$ and $i \in \mathcal{D}$, then $P(j|i) = \beta_{G(j)}$.

Here, the α_k's and β_k's are constants associated with a particular gap pattern k. A sequence evolving according to a substitution process that has these three properties will have constant expected values for the relative frequencies of the $(o + 1)$-mers in $\mathcal{D}$. The first-order partially reversible model, which can be described by a 36×36 matrix, has 170 free parameters, far fewer than the 1260 in the general first-order model.

Partially reversible models are able to capture significantly more information about patterns of selection than the conservation sequence approach used in TWINSCAN 2.0, which considers only patterns of matches, mismatches, and unaligned positions. For example, a first-order partially reversible model can model insertions and deletions separately from base substitutions and can take into account the difference between the rates of transitions and transversions as well as the increased rate of mutation of CpG dinucleotides (Bulmer, 1986). Furthermore, unlike TWINSCAN 2.0, N-SCAN uses a separate conservation model for each codon position in coding sequence, allowing it to model differences in substitution rates between the three positions.

2.6. Conservation score coefficient

Like TWINSCAN, N-SCAN uses log-likelihood scores rather than probabilities internally. The score of a particular column i in the multiple alignment given a state S can be written as

$$\log\left(\frac{T_S(i)}{T_{Null}(i)}\right) + k \cdot \log\left(\frac{C_S(i)}{C_{Null}(i)}\right).$$

Here, T_S and T_{Null} are the target sequence probabilities of the form shown in expression (1) for state S and the null model, respectively. Likewise, C_S and C_{Null} are the conservation model probabilities, as in expression (3). Constant k is an arbitrary constant called the conservation score coefficient which can be used to increase or decrease the impact of the informant sequences on N-SCAN's predictions. To test the effect of the conservation score coefficient on predictive performance, we evaluated gene and exon level accuracy on a human test set consisting of chromosomes 1, 15, 19, 20, 21, and 22 at various values of k. The results are shown in Fig. 2, with TWINSCAN performance inlcuded as a reference. These results

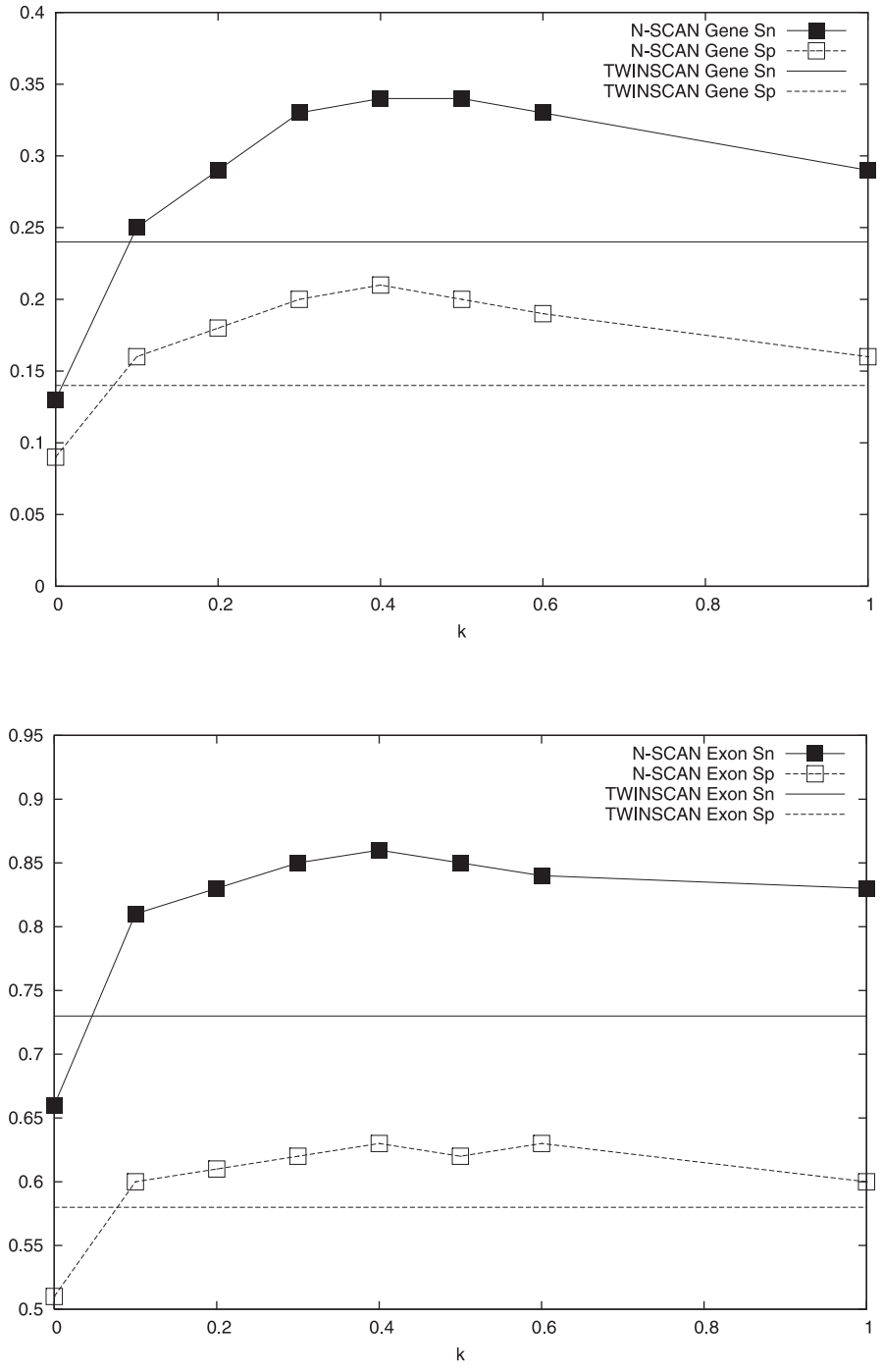

FIG. 2. Exact gene and exon accuracy at different values of the conservation score coefficient.

show that a value of k between 0.3 and 0.6 leads to the best predictive performance. This may be due to the potential of conserved noncoding regions to contribute to a large number of false positive predictions (see below). Importantly, even with k set to 1.0, corresponding to the original model without such a coefficient, N-SCAN performs significantly better than TWINSCAN 2.0.

2.7. State diagram

Figure 3 shows the N-SCAN state diagram. The 5′ UTR and CNS states allow N-SCAN to avoid false positives that would occur if these sequence features were not modeled explicitly. Without these states, conserved noncoding regions would tend to be annotated as coding exons due to their high conservation scores. Instead, N-SCAN tends to annotate conserved regions with a low coding score as CNS. Furthermore, the 5′ UTR states allow N-SCAN to predict exon/intron structure in 5′ UTRs. Simultaneous 5′ UTR and coding region prediction by N-SCAN is discussed in more detail in a paper devoted to the subject (Brown *et al.*, 2005).

Since there was no obvious method for constructing a reliable, general training set for conserved noncoding regions, we used the null target sequence model for the CNS state. This effectively assigns the neutral score of zero to all target sequences under the CNS model. Thus, the score of a putative CNS region is determined entirely by the CNS conservation model, which was estimated from 5′ UTRs. While this resulted in an acceptable model for some types of CNS, more highly conserved CNS was probably not modeled accurately using this method.

2.8. Experimental design

The human gene prediction experiments presented below were performed on the May 2004 build of the human genome, while the fruit fly experiments used the April 2004 build of the *Drosophila melanogaster* genome. They were obtained by downloading the hg17 and dm2 assemblies from the UCSC genome

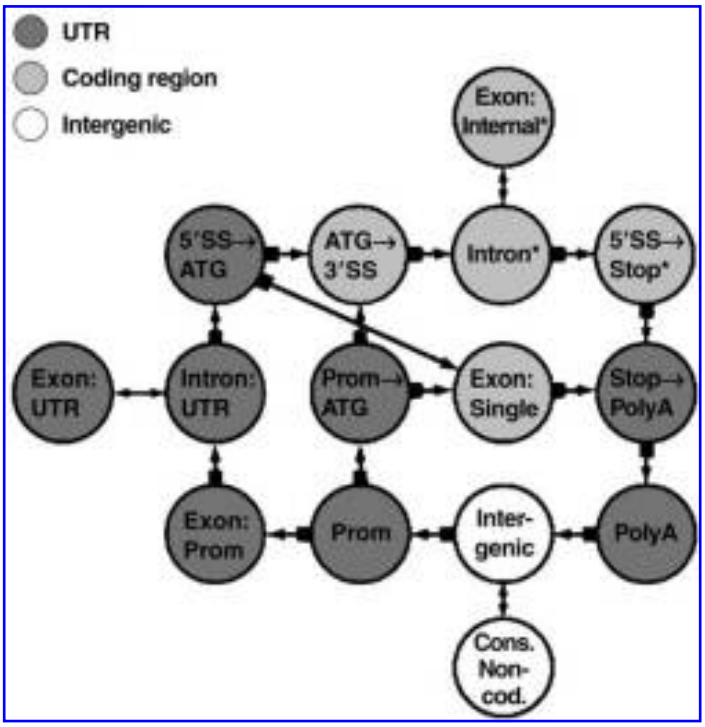

FIG. 3. The N-SCAN state diagram. Intron and exon states with asterisks represent six states each, which are used tracking reading frame and partial stop codons. Only forward strand states are shown; on the reverse strand, all states other than "Intergenic" are duplicated, and initial (ATG → 3′ SS), not terminal (5′ SS → Stop), states track phase and partial stop codons.

browser (Kent *et al.*, 2003). Each group of experiments used a set of annotations consisting of known genes, which was constructed as follows. The human annotation set initially contained the mappings of RefSeqs to the human genome provided by the UCSC genome browser, while the *D. melanogaster* annotation set initially consisted of the FlyBase gene annotations, also provided by the UCSC genome browser. These sets were then filtered to exclude annotations believed likely to have errors. All genes with nonstandard start or stop codons, in-frame stop codons, total coding region length not a multiple of three, donor sites other than GT, GC, or AT, or acceptor sites other than AG or AC were discarded. After this cleaning, the human set contained 16,259 genes and 20,837 transcripts, while the *D. melanogaster* set contained 13,375 genes and 18,769 transcripts.

For the human experiments, N-SCAN used an a whole-genome alignment of human (hg17), chicken (galGal2), mouse (mm5), and rat (rn3) created by MULTIZ (Blanchett *et al.*, 2004). A four-way MULTIZ alignment of *Drosophila melanogaster* (dm2), *Drosophila yakuba* (droYak1), *Drosophila pseudoobscura* (dp2), and *Anopheles gambiae* (anoGam1) was used for the *D. melanogaster* experiments. The alignments were downloaded from the UCSC genome browser. Columns in either alignment with gaps in the target sequence were discarded.

All predictions made by N-SCAN were four-fold cross validated. The first-order partially reversible parameterization was used for all of N-SCAN's Bayesian network CPDs, and N-SCAN's conservation score coefficient was set to 0.4.

3. RESULTS

3.1. Gene prediction performance comparisons

To evaluate the predictive performance of N-SCAN in human, we generated predictions for every chromosome in the hg17 human genome assembly. We then compared the N-SCAN predictions, as well as the predictions of several other *de novo* gene predictors, to our test set of known genes. The gene prediction systems involved in this experiment included one single-genome predictor (GENSCAN), two dual-genome predictors (SGP2 and TWINSCAN 2.0), and two multiple-genome predictors (EXONIPHY and N-SCAN). SGP2 and TWINSCAN 2.0 made use of human–mouse alignments, EXONIPHY used a multiple alignment of human, mouse, rat, and N-SCAN used a multiple alignment of human, mouse, rat, and chicken. The GENSCAN predictions used in this experiment were downloaded from the UCSC genome browser. The SGP2 predictions were downloaded from the SGP2 website. The EXONIPHY predictions were obtained from one of EXONIPHY's creators (A.C. Siepel, personal communication). EXONIPHY does not link exons into gene structures, so its performance at the gene level was not evaluated.

We evaluated both sensitivity and specificity at the gene, exon, and nucleotide levels. Since none of the systems involved in the experiment can predict alternative transcripts, a predicted gene was counted as correct at the gene level if it exactly matched the coding region of any of the transcripts in the test set. Conversely, a set of annotated transcripts at a single gene locus was counted as correctly predicted if any one of them was predicted correctly throughout its coding region. The results of the experiment are shown in Figs. 4, 5, and 6. Note that the specificities are underestimates, since all predicted novel genes—those in the genome but not in the annotation—were counted as incorrect. N-SCAN achieved substantially better performance on both the gene and exon levels than the other four predictors involved in the experiment. On the nucleotide level, N-SCAN had the highest sensitivity, but a lower specificity than EXONIPHY.

We also evaluated the ability of the systems to predict long introns in human genes, a feat notoriously difficult for *de novo* gene predictors (Wang *et al.*, 2003a). The results in Table 1 show that N-SCAN has the greatest sensitivity for each length range we tested. Furthermore, N-SCAN's performance drops off much more slowly with length than that of the other gene predictors. In fact, N-SCAN is able to correctly predict approximately half of the introns in the test set with lengths between 50 kb and 100 kb.

We also performed an experiment to evaluate the performance of N-SCAN in *D. melanogaster*. We generated predictions for the entire euchromatic *D. melanogaster* genome and compared them to our test set of FlyBase genes. We performed the same test using predictions generated by AUGUSTUS (Stanke and Waack, 2003), a single-sequence *de novo* gene predictor. The set of AUGUSTUS predictions was obtained from the AUGUSTUS website. The results of this experiment are shown in Fig. 7. N-SCAN is both more sensitive and more specific at the gene level than AUGUSTUS. At the exon level, N-SCAN

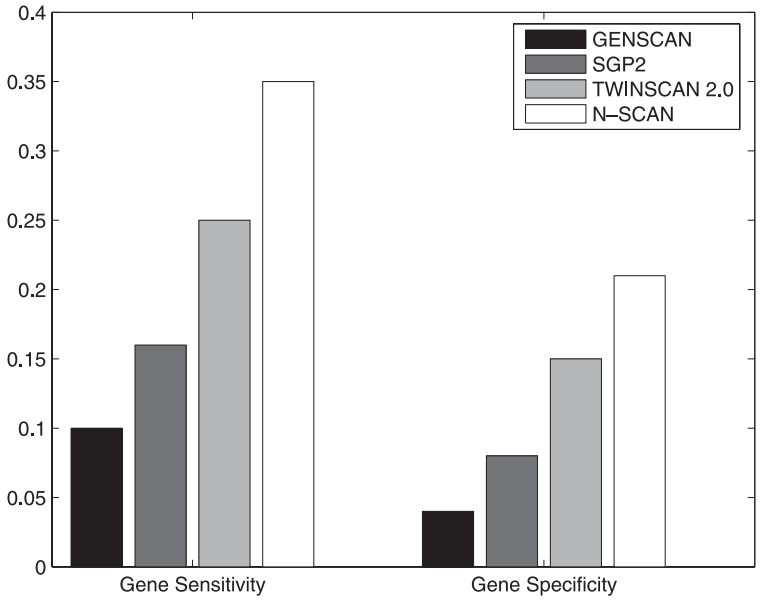

FIG. 4. Exact gene accuracy in human.

achieves a sensitivity 11% higher than AUGUSTUS, at the cost of a 1% decrease in specificity. At the nucleotide level, performance is more balanced: N-SCAN is 7% more sensitive than AUGUSTUS, but 5% less specific.

3.2. Informant effectiveness

To test the effect of multiple informants on N-SCAN's predictive accuracy, we generated four sets of predictions each for human and *D. melanogaster*. The first three sets for each organism use a single informant, while the final set uses all three informants simultaneously. Predictions were generated for human chromosomes 1, 15, 19, 20, 21, and 22, and for the entire euchromatic *D. melanogaster* genome.

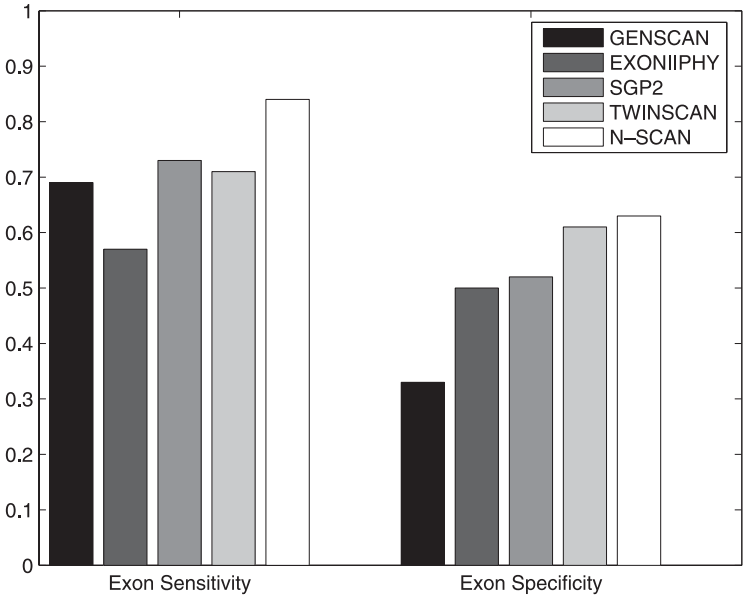

FIG. 5. Exact exon accuracy in human.

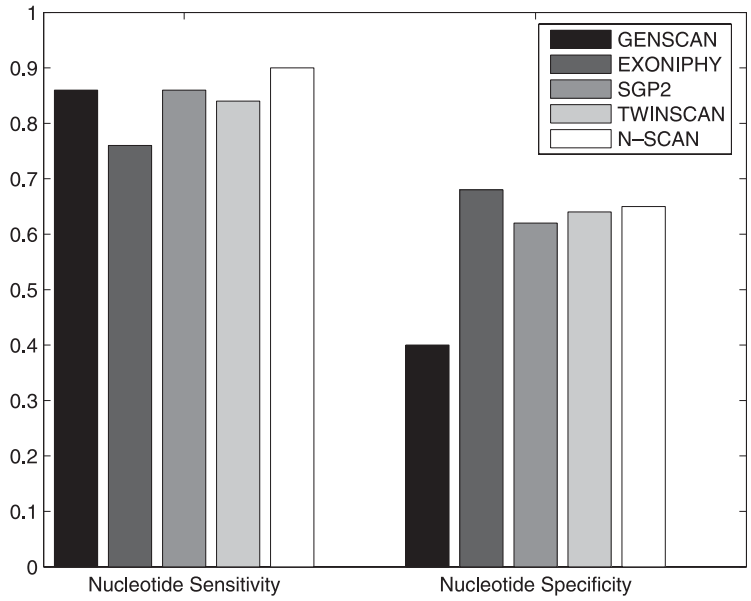

FIG. 6. Nucleotide accuracy in human.

The results of this experiment are shown in Figs. 8 and 9. In *D. melanogaster,* N-SCAN achieved a small but significant boost in performance by using all three informants together. However, in human, using the three informants at once appears to be no better than using mouse alone. In the single informant tests, mouse was the best informant for human gene prediction. This finding is consistent with previous studies of informant effectiveness (Wang *et al.,* 2003b; Zhang *et al.,* 2003).

4. DISCUSSION

We have presented a *de novo* gene prediction system, N-SCAN, which builds on an existing system by incorporating several new features, such as richer substitution models, states for 5′ UTR structure prediction, a conserved noncoding sequence state, and the ability to use information from multiple informant sequences. N-SCAN achieved significantly better genomewide performance on the human and *D. melanogaster* genomes than the other *de novo* gene predictors we tested. Applying N-SCAN to *D. melanogaster* was a simple matter of retraining—no other modifications or tuning were needed.

N-SCAN incorporates information from multiple informant sequences in a novel way which we believe has several potential advantages. First, N-SCAN builds on existing single-sequence models of a target

TABLE 1. INTRON SENSITIVITY BY LENGTH

Length (Kb)	*Count*	*GENSCAN*	*TWINSCAN 2.0*	*N-SCAN*	*SGP2*
0–10	157757	0.68	0.77	**0.86**	0.74
10–20	9519	0.50	0.46	**0.77**	0.69
20–30	3317	0.41	0.22	**0.71**	0.68
30–40	1742	0.30	0.08	**0.64**	0.60
40–50	992	0.28	0.02	**0.64**	0
50–60	652	0.20	0.01	**0.53**	0
60–70	447	0.16	0	**0.52**	0
70–80	314	0.12	0	**0.50**	0
80–90	268	0.10	0	**0.39**	0
90–100	211	0.11	0	**0.48**	0

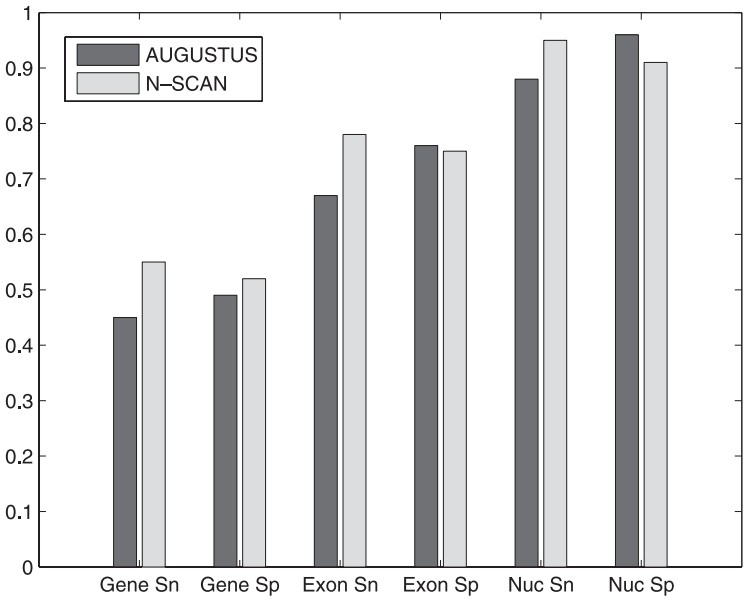

FIG. 7. Accuracy in *D. melanogaster*.

genome. These single-sequence models can be quite sophisticated. For example, the donor splice site model used in GENSCAN and TWINSCAN 2.0 is able to take into account the effect of nonadjacent positions in the splice site signal through the use of a maximal dependence decomposition model. In addition, single-sequence models of a given order generally require fewer parameters than multiple-sequence models of the same order. Therefore, it is possible to use high-order single-sequence models in combination with conservation models of a lower order while maintaining a good fit. In the experiments presented above, some of the N-SCAN target genome models had orders as high as five, while the conservation models were all of order one. Furthermore, because the target sequence is observed, it is possible to obtain a globally optimal estimate of the distributions in the target genome models. The EM estimates for the

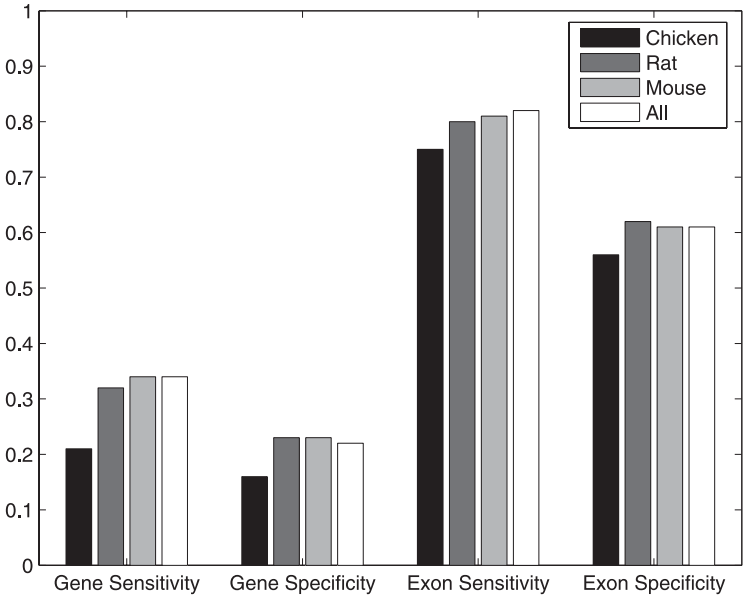

FIG. 8. Informant effectiveness in human gene prediction.

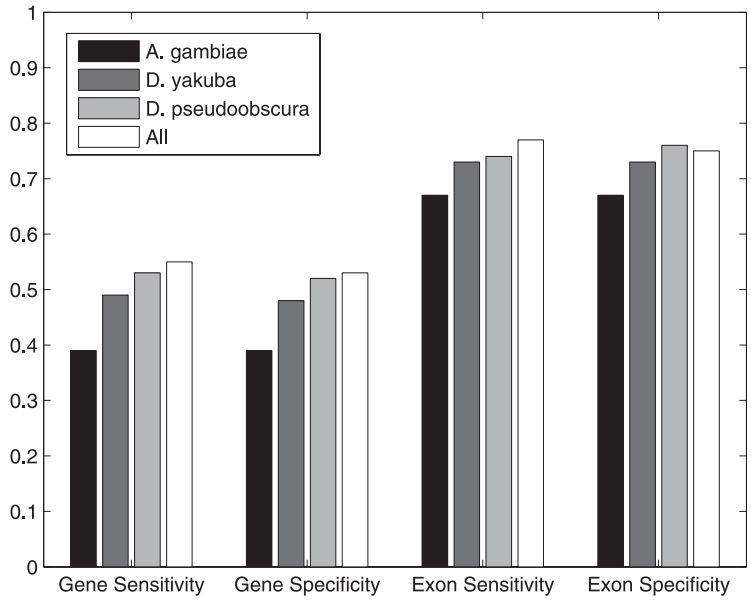

FIG. 9. Informant effectiveness in *D. melanogaster* gene prediction.

conservation models are guaranteed to be only locally optimal and could in principle be far from a global optimum.

Also important is N-SCAN's treatment of gaps and unaligned characters. Instead of treating these characters as missing data, or modeling gap patterns using additional states in the GHMM (Siepel and Haussler, 2004a), N-SCAN deals with them directly in its conservation models. This allows the very significant information they contribute to be taken into account in a natural and efficient way. The price for this ability is that the assumption of a homogeneous Markov substitution process throughout the tree must be abandoned. This assumption is reasonable for models of base mutations between aligned positions in DNA sequences, but is not accurate when considering the nonlinear process of positions becoming unaligned over time. For the sake of illustration, consider an ancestor sequence and a descendant sequence that differ by only a single point mutation. It is not possible for the sequences to have unalignable bases. The alignment will have a gap if the single mutation is an insertion or deletion, but the surrounding regions will provide enough information to align the gap with the right base in the other species. Thus, the instantaneous rates of substitutions leading to unaligned characters are all zero. Yet as divergence increases, a point will be reached where even small changes to the sequence can lead to a whole region becoming unalignable. Therefore, rather than assuming substitutions occur as a result of a homogeneous continuous time Markov process, N-SCAN uses a separate set of parameters to model the substitution probabilities across each branch of the tree. This results in a substantial increase in the total number of parameters for each model, but with appropriate parameterizations, this number is still manageable for context-dependent models of the type presented here.

Relaxing the assumption of a homogeneous substitution process also allows N-SCAN to accurately model alignment columns in which the aligned positions do not share a single functional state. In such a case, patterns of substitution across different branches of the phylogenetic tree are likely to vary significantly, reflecting different evolutionary constraints. This situation cannot be represented by a substitution model that uses the same rate matrix for each branch in the tree. In practice, positions in an alignment column may have different functions as a result of a function-changing mutation, alignment error, or sequencing error. The latter two causes are of particular concern when the alignment contains highly diverged or draft-quality sequences.

The philosophy behind N-SCAN, then, is to model alignments as they are, rather than modeling the process of molecular evolution directly. Genome sequence alignments are artifacts manufactured by humans in an attempt to represent both the genome sequences of contemporary species and their descent. It is remarkable that we can construct such representations at all, but they are imperfect models with many

idiosyncratic properties resulting from errors in sequencing, assembly, and alignment. Furthermore, the frequency of such errors is not constant throughout the evolutionary tree. It depends on factors that are unrelated to evolution, such as depth of sequencing coverage. To ignore these properties and treat genome sequence alignments as veridical is a risky and potentially fragile approach. The N-SCAN approach is expected to be more robust to such errors and variations in error rate.

Although its accuracy is high, N-SCAN raises fundamental questions that will have to be addressed in future studies. Among these are the following.

1. Why must we "dial down" the influence of the conservation model in order to get optimal performance?
2. Since using multiple informant genomes seems to be no better than using mouse alone, why does N-SCAN perform so much better on the human genome than previous systems?
3. Why has no one yet succeeded in predicting mammalian gene structures more accurately with multiple informant genomes than with just one?

One conjecture about question 1 is that our models of conserved sequence do not cover enough of the true sources of conservation, so too much conserved sequence gets thrown into coding exons, UTRs, splice sites, and other sources that we do model. We plan to investigate this further by examining the types of errors N-SCAN makes when the conservation score coefficient is eliminated.

As for question 2, likely explanations include the fact that N-SCAN's nucleotide substitution models are specific to codon position and to nucleotide pairs, whereas TWINSCAN 2.0 treats all mismatches as equal. N-SCAN's state for conserved intergenic sequence may also contribute to its accuracy.

Question 3 presents the greatest mystery. The mouse, rat, and chicken genomes may have some particular characteristics that explain why they do not yield better performance than mouse alone. Specifically, only the mouse genome sequence is in really good condition (in finishing). While the rat is a close second, its divergence from mouse is too little to add much information. The chicken genome, on the other hand, is much more diverged from human, making it more difficult to align accurately to human and less predictive of function in human. Thus, perhaps finishing the dog, cow, or lemur and combining it with mouse would lead to an improvement over mouse alone. On the other hand, the answer may not lie in the sequences but in the model. Perhaps if we were able to model more sources of sequence conservation, such as RNA genes, transcription factor binding sites, and micro-RNA targets, we would be able to better isolate the coding-related information that additional genomes should provide.

ACKNOWLEDGMENTS

We thank Mikhail Velikanov for providing the set of filtered RefSeq genes used for training and evaluation. We also thank Adam Siepel for converting the EXONIPHY predictions on hg16 to hg17 coordinates and for an enlightening discussion. Finally, thanks to all the members of the Brent lab who developed and maintained the TWINSCAN code base, from which N-SCAN was developed. This work was supported by grant HG02278 from the NIH to M.R.B.

REFERENCES

Alexandersson, M., Cawley, S., and Pachter, L. 2003. SLAM: Cross-species gene finding and alignment with a generalized pair hiddden Markov model. *Genome Res.* 13, 496–502.

Blanchett, M., *et al.* 2004. Aligning multiple genomic sequences with the threaded blockset aligner. *Genome Res.* 14, 708–715.

Brown, R.H., Gross, S.S., and Brent, M.R. 2005. Begin at the beginning: Predicting genes with 5′ UTRs. *Genome Res.* 15, 742–747.

Bulmer, M. 1986. Neighboring base effects on substitution rates in pseudogenes. *Mol. Biol. Evol.* 3, 322–329.

Burge, C., and Karlin, S. 1997. Prediction of complete gene structures in human genomic DNA. *J. Mol. Biol.* 268, 78–94.

Felsenstein, J. 1982. Evolutionary trees from DNA sequences. *J. Mol. Evol.* 17, 368–376.

Flicek, P., Keibler, E., Hu, P., Korf, I., and Brent, M.R. 2003. Leveraging the mouse genome for gene prediction in human. *Genome Res.* 13, 46–54.

Gerhard, D., *et al.* 2004. The status, quality, and expansion of the NIH full-length cDNA project: The mammalian gene collection (MGC). *Genome Res.* 14, 2121–2127.

Kent, W., Sugnet, C., Furey, T., Roskin, K., Pringle, T., Zahler, A., and Haussler, D. 2003. The human genome browser at UCSC. *Genome Res.* 12, 996–1006.

Korf, I., Flicek, P., Duan, D., and Brent, M.R. 2001. Integrating genomic homology into gene structure prediction. *Bioinformatics* 17(Suppl. 1), S140–148.

Lió, P., and Goldman, N. 1998. Models of molecular evolution and phylogeny. *Genome Res.* 8, 1233–1244.

McAuliffe, J., Pachter, L., and Jordan, M.I. 2003. *Multiple-sequence functional annotation and the generalized hidden Markov phylogeny*. Technical report 647, Department of Statistics, University of California, Berkeley.

Parra, G., Agarwal, P., Abril, J., Wiehe, T., Fickett, J., and Guigo, R. 2003. Comparative gene prediction in human and mouse. *Genome Res.* 13, 108–117.

Pedersen, J., and Hein, J. 2003. Gene finding with a hidden Markov model of genome structure and evolution. *Bioinformatics* 19, 219–227.

Siepel, A.C., and Haussler, D. 2004a. Computational identification of evolutionarily conserved exons. *Proc. 8th Ann. Conf. on Research in Computational Molecular Biology (RECOMB '04).*

Siepel, A.C., and Haussler, D. 2004b. Phylogenetic estimation of context-dependent substitution rates by maximum likelihood. *Mol. Biol. Evol.* 21, 448–468.

Stanke, M., and Waack, S. 2003. Gene prediction with a hidden Markov model and a new intron submodel. *Bioinformatics* 19(Suppl. 2), ii215–ii225.

Tenney, A.E., Brown, R.H., Vaske, C., Lodge, J., Doering, T.L., and Brent, M.R. 2004. Gene prediction and verification in a compact genome with many small introns. *Genome Res.* 14, 2330–2335.

Wang, J., *et al.* 2003a. Vertebrate gene prediction and the problem of large genes. *Nature Rev. Genet.* 4(9), 741–749.

Wang, M., Buhler, J., and Brent, M.R. 2003b. The effects of evolutionary distance on TWINSCAN, an algorithm for pair-wise comparative gene prediction. *Cold Spring Harbor Symp. Quant. Biol* 68, 125–130.

Waterston, R.H., *et al.* 2002. Initial sequencing and comparitive analysis of the mouse genome. *Nature* 420, 520–562.

Zhang, L., Pavlovic, V., Cantor, C., and Kasif, S. 2003. Human–mouse gene identification by comparative evidence integration and evolutionary analysis. *Genome Res.* 13, 1190–1202.

Address correspondence to:
Samuel S. Gross
Room S260
James H. Clark Center
318 Campus Drive
Stanford University
Stanford, CA 94305

E-mail: ssgross@cs.stanford.edu