Application of Hidden Markov Models to Gene Prediction in DNA

Michael M. Yin
Department of Computer Science
New Jersey Institute of Technology
University Heights, Newark, NJ 07102, USA
yin@homer.njit.edu

Jason T. L. Wang
Department of Computer Science
New Jersey Institute of Technology
University Heights, Newark, NJ 07102, USA
jason@cis.njit.edu

Abstract

Programs currently available for gene prediction from within genomic DNA are far from being powerful enough to elucidate the gene structure completely. In this paper, we develop a hidden Markov model (HMM) to represent the degeneracy features of splicing junction donor sites in eucaryotic genes. The HMM system is fully trained using an expectation maximization algorithm and the system performance is evaluated using the 10-way cross-validation method. Experimental results show that our HMM system can correctly classify more than 95% of the candidate sequences into the right categories. More than 91% of the true donor sites and 97% of the false donor sites in the test data are classified correctly. These results are very promising considering that only the local information in DNA is used. This model will be a very important component of effective and accurate gene structure detection system currently being developed in our lab.

1. Introduction

Identification or prediction of transcribed sequences from within genomic regions has been a major rate-limiting step in the pursuit of genes. The bioinformatics approach for gene detection means using computer programs to elucidate gene structure from DNA sequence signals, including start codon, splicing junction donor sites and acceptor sites, stop codon, etc. Since 1990s, a number of programs have been developed and are currently available for gene coding region localization. But the DNA sequence signals involved in gene determination are usually ill defined, degenerate and highly unspecific. Given the current detection methods it is usually impossible to distinguish the signals truly processed by the cellular machinery from those that are apparently non-functional [6].

Recently, Roderic Guigo extended the evaluation carried out by Burset and Guigo [5] concerning the accuracy of a number of gene structure prediction programs. These authors tested the programs on a large set of vertebrate sequences with simple gene structure. It was found that even for the programs receiving an average rate of 37% to 76% for exon sensitivity and specificity, they are able to elucidate gene structure in only about 5% to 40% [6]. So Guigo claimed that those programs perform rather poorly when confronted with the problem of fully elucidating gene structure. In fact, such programs incorporate little knowledge of the biological processes underlying gene specification. They rely mainly on sequence coding statistics, a consequence of the existence of the genes [6].

These research results strongly suggest that incorporating a model of the biological events involved in the specification for the genes (transcription, splicing and translation) would benefit the computational methods for gene structure prediction. Our research is targeted toward developing more effective and accurate methods for identifying gene structures. Splicing junction donor and receptor sites are the most important functional gene structure signals. Earlier we have developed a donor Motif model [13] and have used pattern-matching techniques [10, 11, 12] for donor classification. In this paper, we introduce a hidden Markov model (HMM) to represent the degeneracy features of the splicing sites. We develop an EM-like algorithm to train our HMM system. Then we use the 10-way cross-validation method to evaluate our system for classifying donor sites in unlabeled test DNA sequences.

Hidden Markov models (HMMs) have been used extensively to describe sequential data or processes such as speech recognition. Researchers in computational biology have recently started to use HMMs for biological sequence analysis. Lukashin, Borodovsky [8] and their colleagues [4] successfully applied HMMs to the detection of protein coding regions in procaryote. Audic and Claverie [2] reported their using of Markov transition matrices to detect eucaryotic promoters. Salzberg's group at Johns Hopkins University, Baltimore, developed an HMM system, called VEIL (Viterbi Exon-Intron

Locator), for finding eucaryotic genes [7]. Our approach differs from VEIL by using a different topology of HMM and by employing two modules in the HMM donor model: one for true donor sites, and the other for false donor sites.

There are also many other pioneers in this field. Even though the current systems are far from being powerful enough for gene structure elucidation, the information these researchers provide is valuable, and research on automated gene detection using HMM is of great potentiality.

2. Our Approach

2.1. Donor HMM models

Donor sites are conserved boundary sequences at the 5' splicing sites in DNA. The conserved sequences include 9 nucleotide bases with GT (GU in mRNA) almost invariable to all donor sites [1]. An example of a donor site sequence looks like the following:

CAGGTGAGT

The nucleotide G occurs at position 4 and nucleotide T occurs at position 5 in a donor site. We refer to a 9-base donor that exists in a real gene sequence as a “true donor site.” Note that in all true donor sites, G and T occur at position 4 and position 5, respectively. We refer to a 9-base non-donor sequence in which the G and T also occur at position 4 and position 5 as a “false donor site”. Notice that we do not consider those sequences without G, T being at position 4 and position 5, because they are deemed to be non-donor sequences. Given a 9-base sequence with the G, T being at position 4 and position 5, referred to as a “candidate sequence”, our algorithm tries to determine whether the candidate is a true donor site or a false donor site. The donor HMM models are designed so that they can capture the consensus and degenerate properties occurring in true donor sites.

An HMM with 9 states and a set of transitions is defined for modeling a true donor site, represented as a digraph where states correspond to vertices and transitions to edges. At each state, the HMM will generate a base *b* in {A, G, C, T} according to the state and transition probabilities, with the exception of state 4 and state 5. At state 4, the HMM constantly generates base *b* = G, and at state 5, the HMM constantly generates *b* = T. Each state *s* is associated with a discrete output probability distribution, *P(s)*. We can easily see that, for state 4 and state 5, *P(s) = 1*. Except at state 3 and state 4, each base b at a state has four possible transitions to the next state. Each transition has a probability, *P(t)*, which represents the probability that the HMM makes that transition. Each base at state 3 has a fixed transition, namely one to the base G at state 4. Similarly, at state 4, the base **G** has a fixed transition, namely one to base **T** at state 5. Figure 1 shows the graph model of the donor HMM.

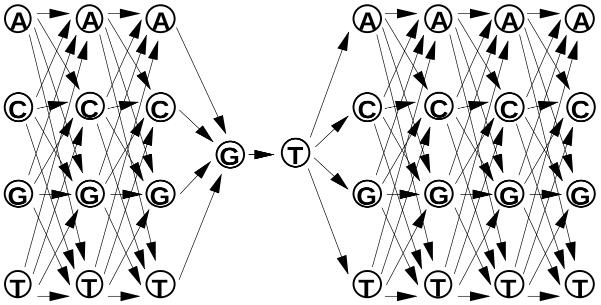

Figure 1. The donor HMM. There are 9 states in this model. Except state 4 and state 5, there are four possible bases at each state, and a base at one state may have four possible ways to transit to next state. States 4 and 5 are constant, and the transition from state 4 to state 5 is also constant with a probability of 1. In a gene sequence, states 1 through 3 belong to the exon and states 4 to 9 are part of the intron.

2.2. Training algorithm

In order to detect or classify true donor sites from a set of candidate sequences, we build two modules from the above HMM. One is called the True Donor Module, and the other is called the False Donor Module. We train the True Donor Module using a training data set containing all true donor site sequences, and train the False Donor Module using a training data set containing all false donor site sequences. A given candidate sequence will be tested by these two modules. Let S_{cand} be a candidate sequence. $M^{(t)}$ stands for the True Donor Module and $M^{(t)}$ stands for the False Donor Module. Let $P(Y=1 \mid S_{cand} , M^{(t)})$ be the probability of S_{cand} being a true donor site when tested by the True Donor Module.

Let $P(Y=1 \mid S_{cand} , M^{(t)})$ be the probability of S_{cand} being a false donor site when tested by the False Donor Module. According to our HMM donor model, the only transition probability from state 4 to state 5 (G$\rightarrow$T) can be 100%. Any other transition probabilities can not be 100%. For a donor site with 8 transition stages, its probability can never reach 100%. For a candidate sequence S_{cand}, we first assume it is a true donor site, and pass it through the True Donor Module. We obtain $P(Y=1 \mid S_{cand}, M^{(t)})$, the probability of this candidate being a true donor site. Then we assume S_{cand} to be a false donor site and pass it through the False Donor Module. We get $P(Y=0 \mid S_{cand}, M^{(f)})$, the probability of S_{cand} being a false donor site. Comparing these two values, we assign a score to $S_{cand.}$

This candidate sequence will be assigned to the true donor or false donor category depending on the score it obtains.

To train these HMM modules, we use a method derived from the expectation maximization (EM) algorithm. The original EM method takes as input a set of unaligned sequences and a motif [3] sequences with the same length, and all these sequences can be aligned to each other, we design an EM like algorithm working as follows.

The HMM module topology is fixed, and all of the transition probabilities and state probabilities are initialized to 0. Then we input the first subset of positive training data (e.g. 100 sequences) to the True Donor Module, record the number of the individual base at each state, and the number of individual transitions from a state to the next. We then compute the prior probabilities for all the states and transitions in the True Donor Module. Upon obtaining the prior probabilities, we input another subset of positive training data to the True Donor Module, and all the posterior probabilities are re-adjusted. We then calculate the differences, *diff*, for all the probabilities between the prior and posterior probabilities. If some of the *diff* are larger than a predefined threshold value ε, set the current posterior probabilities as the new prior probabilities, and the new data set is then run through the True Donor Module again and the probabilities are further refined. This training process is iterated until the changes in all probabilities in the True Donor Module are smaller than ε. This algorithm is guaranteed to get a locally maximized expectation for all the probabilities in the HMM model. The False Donor Module is trained using the negative training data in the same way as for the True Donor Module.

2.3. Classification algorithm

We define $ftr_i^t(b_{i-1}, b_i)$ as the probability of a transition from base b_{i-1} to b_i using the HMM True Donor Module. We define a flag variable Y to be 1 if a sequence belongs to the true donor category and 0 otherwise. Let $P(S_{cand} \mid Y=1,M^{(t)})$ be the probability of the candidate sequence, S_{cand} given that it is a true donor site. This can be written as:

$$P(S_{cand} \mid Y=1, M^{(t)}) = \prod_{i=1}^{8} ftr_i^{(t)}(b_{i-1}, b_i), \quad b \ni (a, g, c, t)$$

As defined before, $P(Y=1 \mid S_{cand}, M^{(t)})$ is the probability of a true donor site given a candidate sequence, S_{cand}. According to Bayes' law:

$$P(A \mid B) = \frac{P(B \mid A)P(A)}{P(B)}$$

we can compute $P(Y=1 \mid S_{cand} , M^{(t)})$ by the equation bellow:

$$P(Y=1 \mid S_{cand}, M^{(t)}) = \frac{P(S_{cand} \mid Y=1,\ M^{(t)})P(Y=1)}{P(S_{cand})}$$

When comparing a set of patterns to detect true donor sites, we can treat the underlying prior $P(Y=1)$ as a constant [9]. $P(S_{cand})$ is the product of the individual base probabilities for b_0, b_1, …, b_8 in the candidate sequence:

$$P(S_{cand}) = \prod_{i=0}^{8} P(b_j \mid Y_j = 1, M^{(t)}), \quad b \ni (a, g, c, t)$$

Similarly, we use the HMM False Donor Module to compute $P(S_{cand} \mid Y=1,M^{(t)})$, the probability of false donor site given a candidate sequence S_{cand}. So, we can write the false donor site counterparts of the above equations:

$$P(S_{cand} \mid Y=0, M^{(f)}) = \prod_{i=1}^{8} ftr_i^{(f)}(b_{i-1}, b_i), \quad b \ni (a, g, c, t)$$

$$P(Y=0 \mid S_{cand}, M^{(f)}) = \frac{P(S_{cand} \mid Y=0,\ M^{(f)})P(Y=0)}{P(S_{cand})}$$

$$P(S_{cand}) = \prod_{i=0}^{8} P(b_j \mid Y_j = 0, M^{(f)}), \quad b \ni (a, g, c, t)$$

As described earlier, for a candidate sequence S_{cand}, we always compute its probability being a true donor site and its probability being a false donor site. If S_{cand} gets p_1 of 0.7 through the True Donor Module and gets p_2 of 0.3 through the False Donor Module, we predict it to be a true donor site. It should be pointed out that (p_1 + p_2) may not be 1. Even we predict S_{cand} to be a true donor site based on p_1 and p_2, S_{cand} may not be a “real” donor site. This is where the prediction error comes from.

Given a candidate sequence S_{cand} and a trained HMM with a True Donor Module and a False Donor Module in it, our algorithm will find the two most likely sets of states through the two modules for S_{cand}. Then the algorithm calculates $P(S_{cand} \mid Y=1,M^{(t)})$ and $P(S_{cand} \mid Y=0, M^{(f)})$ according to the HMM output. A score is given to the candidate sequence using the scoring function below:

$$\frac{P(S_{cand} \mid Y=1, M^{(t)})}{P(S_{cand} \mid Y=0, M^{(f)})}$$

We compare this score with an experimentally defined threshold, to assign S_{cand} to the true donor or false donor category.

3. Experiments and Results

We applied the 10-way cross-validation method to evaluate how well the HMM system would perform when tested on data that are not in the training data set. A cross validation is a standard experimental technique for determining how well a classification system will perform on unseen data [7]. Specifically, we randomly partitioned our test data set into 10 subsets of as nearly equal size as possible. For each such subset *S*, the HMM system is trained using the other nine subsets (*i.e.*, all sequences excluding *S* are used as the training data). The system is then tested on the sequences in *S* (thus, the training data consisted of 90% and the test data consisted of 10% of the sequences). In our experiments, the sequences at hand included 850 DNA fragments from real genes. Part of these DNA fragments were obtained by anonymous FTP from: *ftp.ics.uci.edu/pub/machine-learning-database*. The rest of them were obtained from the set of genes originally collected by Burset and Guigo [5].

In our experiments, we randomly partitioned the sequences at hand into 10 subsets, each having 85 DNA fragments. In each run, we refer to the 9 subsets of the DNA sequences (765 fragments) used for training as *training data* and the other 85 DNA fragments as *test data*. In each run, the system was trained using the train date and tested on the test data.

Table 1 shows the final state transition probabilities for the True Donor Module in one of the ten copies of the HMM system. The state transition probabilities in the same copy of the system for the False Donor Module are shown in Table 2. Comparing the transition probabilities in Table 1 and Table 2, we can see that our HMM system maximized the differences between true donor sites and false donor sites. From the data in Table 1 and the individual nucleotide frequency at each state in the True Donor Module (data is not shown), we can derive the consensus sequence for the true donor site as shown in Figure 2. This consensus sequence for the true donor site is the same as reported before [1, 13].

Table 1. State transition probabilities for the True Donor Module

	$ftr_i^t(b_{i-1}, b_i)$							
Transition	**(b_0, b_1)**	**(b_1, b_2)**	**(b_2, b_3)**	**(b_3, b_4)**	**(b_4, b_5)**	**(b_5, b_6)**	**(b_6, b_7)**	**(b_7, b_8)**
A→A	0.209	0.044	0.000	0.000	0.000	0.331	0.025	0.005
A→G	0.048	0.0473	0.098	0.000	0.000	0.075	0.633	0.014
A→C	0.030	0.013	0.000	0.000	0.000	0.051	0.024	0.009
A→T	0.031	0.035	0.000	0.000	0.000	0.051	0.027	0.010
G→A	0.101	0.025	0.000	0.000	0.000	0.357	0.004	0.122
G→G	0.047	0.124	0.771	0.000	0.000	0.041	0.107	0.199
G→C	0.017	0.008	0.000	0.000	0.000	0.051	0.005	0.127
G→T	0.010	0.017	0.000	1.000	0.000	0.008	0.004	0.404
C→A	0.238	0.029	0.000	0.000	0.000	0.017	0.008	0.033
C→G	0.042	0.067	0.039	0.000	0.000	0.001	0.058	0.000
C→C	0.056	0.017	0.000	0.000	0.000	0.004	0.033	0.007
C→T	0.051	0.025	0.000	0.000	0.000	0.003	0.010	0.024
T→A	0.018	0.000	0.000	0.000	0.507	0.004	0.003	0.008
T→G	0.037	0.107	0.092	0.000	0.456	0.004	0.054	0.025
T→C	0.034	0.001	0.000	0.000	0.025	0.003	0.001	0.003
T→T	0.030	0.013	0.000	0.000	0.012	0.001	0.005	0.012

Table 2. State transition probabilities for the False Donor Module

	$ftr_i^{\,f}\,(b_{i-1}, b_i)$							
Transition	**(b_0, b_1)**	**(b_1, b_2)**	**(b_2, b_3)**	**(b_3, b_4)**	**(b_4, b_5)**	**(b_5, b_6)**	**(b_6, b_7)**	**(b_7, b_8)**
A→A	0.039	0.047	0.000	0.000	0.000	0.025	0.052	0.060
A→G	0.064	0.054	0.219	0.000	0.000	0.027	0.038	0.070
A→C	0.032	0.023	0.000	0.000	0.000	0.036	0.037	0.041
A→T	0.017	0.079	0.000	0.000	0.000	0.051	0.046	0.048
G→A	0.077	0.072	0.000	0.000	0.000	0.067	0.086	0.046
G→G	0.136	0.161	0.328	0.000	0.000	0.199	0.113	0.135
G→C	0.088	0.026	0.000	0.000	0.000	0.087	0.068	0.042
G→T	0.059	0.074	0.000	1.000	0.000	0.080	0.057	0.047
C→A	0.060	0.078	0.000	0.000	0.000	0.063	0.058	0.068
C→G	0.046	0.017	0.128	0.000	0.000	0.024	0.023	0.037
C→C	0.078	0.041	0.000	0.000	0.000	0.070	0.074	0.083
C→T	0.060	0.114	0.000	0.000	0.000	0.095	0.091	0.065
T→A	0.027	0.022	0.000	0.000	0.101	0.019	0.024	0.034
T→G	0.090	0.096	0.326	0.000	0.433	0.074	0.097	0.092
T→C	0.047	0.035	0.000	0.000	0.252	0.056	0.076	0.063
T→T	0.028	0.059	0.000	0.000	0.214	0.064	0.060	0.066

Position	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**
Nucleotide	**C or A**	**A**	**G**	**G**	**T**	**T or A**	**A**	**G**	**T**

Figure 2. Consensus sequence for the true donor site. This sequence is derived according to the state transition probabilities and state probabilities in the True Donor Module.

3.1. Performance evaluation

As stated in Section 2, we treat all 9-base sequences with the G, T being at position 4 and position 5 as “candidate sequences”. Our algorithm tries to determine whether a candidate sequence is a true donor site or a false donor site. Each DNA fragment in our test data set contains exactly one true donor site, but possibly several false donor sites. One of the most important steps in detecting gene structure is to predict the true donor sites correctly in DNA sequences. This means that we need to correctly classify all the candidate sequences and assign them to the true or false donor category respectively. Table 3 shows the total number of candidate sequences (*Candidate*), the total number of true donor sites (*True*) and the total number of false donor sites (F*alse*) in each set of the test data.

Table 3. Test data in different subsets

Sub set	***Candidate***	***True***	***False***
1	292	85	207
2	279	85	194
3	285	85	200
4	238	85	153
5	284	85	199
6	270	85	185
7	269	85	184
8	293	85	208
9	268	85	183
10	276	85	191

Table 4 shows how many candidate sequences were correctly classified and how many were mis-classified in each set of the test data. Here we use Sensitivity (Sn) to evaluate the performance of our approach. Sensitivity is the proportion of candidate sequences (C.S.) in the test data that are correctly classified. Sensitivity is calculated by the formula below:

$$S_n = \frac{Total\ C.S.\ that\ \ Are\ Correctly\ Classified}{Total\ C.S.}$$

Table 4. Results for donor classification

Sub set	*Candidate*	*Correctly Classified*	*Mis-Classified*	S_n
1	292	277	15	0.949
2	279	262	17	0.939
3	285	271	14	0.951
4	238	226	12	0.950
5	284	275	9	0.968
6	270	264	6	0.978
7	269	255	14	0.948
8	293	278	15	0.949
9	268	258	10	0.963
10	276	263	13	0.953
Average				0.955

The results in Table 4 show that, in our 10-way cross-validation experiment, the system can correctly classify 95.5% of the input test data on average. These results are for the overall classification of the test data. Included in the mis-classified data are candidate sequences that are true donor sites but classified as false (false negative, *FN*), and that are false donor sites but classified as true (false positive, *FP*). It is very important to estimate how well the HMM system would perform for true donor sites. We use Sensitivity (S_n^{true}) and Specificity (S_p^{true}) to evaluate the performance of the HMM system for true donor sites. S_n^{true} is the ratio between correctly classified true donor sites (TP) and the total number of true donor sites in the test data (*TP* + *TN*). S_p^{true} is the ratio between correctly classified true donor sites (*TP*) and the total number of true donor sites classified (*TP* + *FP*). Table 5 shows the results where S_n^{true} and S_p^{true} were calculated using the following formulas:

$$S_n^{true} = \frac{TP}{TP + FN}$$

$$S_p^{true} = \frac{TP}{TP + FP}$$

From Table 5, we can see that, on average, our HMM system can correctly detect 91.3% of the true donor sites in the test data set, and about 94% of the true donor sites the system classified are correct. We also did similar calculations to evaluate the performance of the system for classifying the false donor sites and the results are shown in Table 6. Sensitivity for false donor sites (S_n^{false}) is the ratio between correctly classified false donor sites (*TF*) and the total number of false donor sites in the test data (*TF* + *FP*):

$$S_n^{false} = \frac{TF}{TF + FP}$$

Table 5. Performance Evaluation for detecting true donor sites

Sub set	**TP**	**FP**	**FN**	S_n^{true}	S_p^{true}
1	77	7	8	0.906	0.917
2	76	8	9	0.894	0.905
3	74	3	11	0.871	0.961
4	77	4	8	0.906	0.951
5	78	2	7	0.918	0.975
6	84	5	1	0.988	0.944
7	76	5	9	0.894	0.938
8	74	4	11	0.871	0.949
9	81	6	4	0.953	0.931
10	81	7	6	0.929	0.919
Average				0.913	0.939

Specificity for false donor sites (S_p^{false}) is the ratio between correctly classified false donor sites (*TF*) and the total number of false donor sites classified (*TF* + *FN*):

$$S_p^{false} = \frac{TF}{TF + FN}$$

On average, the system can correctly classify more then 97% of the total false donor sites in the sequences at hand, and about 96% of the false donor sites classified are correct.

4. Discussion

To investigate how well the HMM system can discriminate true donor sites from false donor sites when a group of candidate sequences is presented to the system, we randomly selected two copies of the scores from our 10-way cross-validation experiment and plotted in Figure 3. A striking difference can be observed by comparing the curves in Figure 3. The scores for true donor sites can be higher than 330000. The highest score for the false donor sites is less than 19. The scores for the false donor sites can be lower than 0.000934. And the lowest score for true donor sites is about 0.1. There are 7 sequences in 170 true donor sites with scores lower then 5, and there are only 4 sequences in 170 false donor sites with scores higher than 5[1].

Table 6. Performance evaluation for detecting false donor sites

Sub set	TF	FP	FN	S_n^{false}	S_p^{false}
1	200	7	8	0.966	0.962
2	186	8	9	0.959	0.954
3	197	3	11	0.985	0.947
4	149	4	8	0.974	0.949
5	197	2	7	0.990	0.966
6	180	5	1	0.973	0.994
7	179	5	9	0.973	0.952
8	204	4	11	0.981	0.949
9	177	6	4	0.967	0.978
10	184	7	6	0.963	0.968
Average				0.973	0.962

Figure 4 shows the score distribution of true donor sites and false donor sites. These data are the same as in Figure 3 and processed in a different way. The true donor sites are grouped into subgroups according to the ceiling of the logarithmic values of their scores. For example, if a donor site obtains a score 350 and another donor site gets a score 850, they will be put into the same subgroup with a donor site that has a score of 1000, because the ceilings of the logarithmic values of these scores are all 3. The false donor sites are grouped into subgroups in the same way.

The curves in Figure 4 indicate that about 70% of the true donor sites are scored between 100 and 100000 by the HMM system, and about 70% of the false donor sites are assigned scores between 0.1 and 0.0001. There are about 4% of the true donor sites with scores lower than 5. These are the true donor sites overlapped with the false donor sites and are the source of false negative result (FN). Less than 3% the false donor sites are of scores equal to or greater than 5, and these contribute to the false positive (FP) results. The results presented above demonstrate that the Hidden Markov Model we developed can be used to discover the degenerate pattern features of the splicing junctions to a great degree. Based on this model, our system can correctly classify more than 95\% of the candidate sequences into the right categories (true or false donor site categories). More than 91% of the true donor sites in the test data are classified correctly, and more than 94% of the true donor sites detected by the system are correctly recognized. For the false donor sites, more than 97% of them in the test data are classified as false, and 96% of the false donor sites recognized by the system are correct.

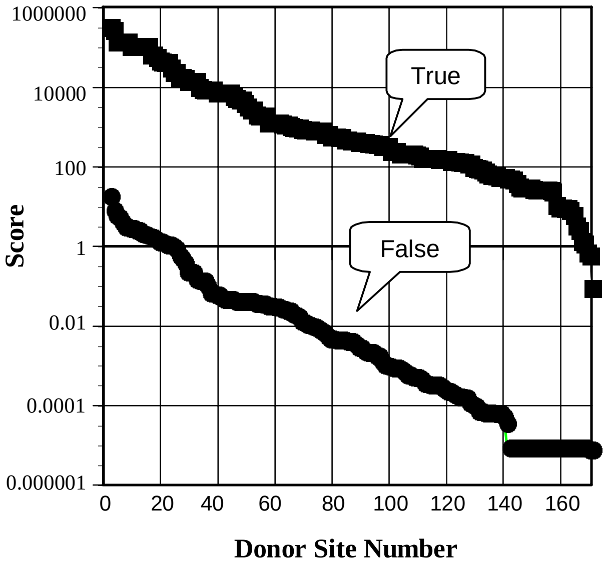

Figure 3. Test scores assigned to the true and false donor sites by the HMM system. Donor sites in each category are sorted according to their scores. The callout indicates whether a curve is for true donor or false donor sites.

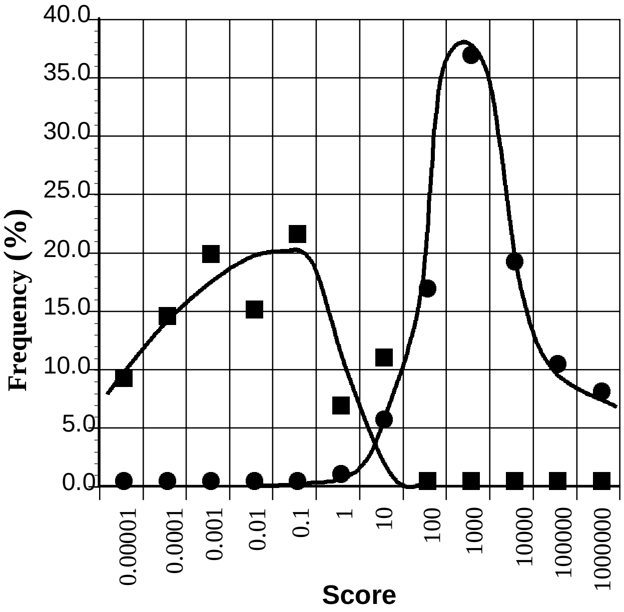

Figure 4. Score distributions for donor sites. See text for the method to group the donor sites in each category according to their scores. Solid circles: true donor sites, Solid squares: false donor sites.

[1] 5 is the threshold that the system used for donor classification.

5. Conclusion

In this paper, we have develop a hidden Markov model (HMM) to represent the degeneracy features of splicing junction donor sites in eucaryotic genes. The HMM system was fully trained using an expectation maximization (EM) algorithm and the system performance was evaluated using the 10-way cross-validation method.

Recently, Salzberg's group at Johns Hopkins University, Baltimore, also developed an HMM system, called VEIL (Viterbi Exon-Intron Locator), for finding eucaryotic genes [7]. In VEIL system, there are nine states in the donor site model. Each state can output only one base out of four bases. Each base at a state has four possible edges to the bases at the next state. The above holds for all nine states in the model. In contrast, our HMM donor model differs from VEIL in several ways. First, in our HMM donor model, state 4 can output only base **G**, and state 5 can output only base **T**. Second, the bases in state 3 and state 4 can only have one possible edge to the next state. Third, there are two modules in our HMM donor model: one for true donor sites, and the other for false donor sites. We think our HMM donor model is better for capturing the consensus and degenerate properties occurring in true donor sites.

It is worth to point out that we only used the local information in this research for the donor classification. When combining our HMM system with the global gene structure information, it is likely that one can achieve even better results for site recognition. This is the research underway in our group. Currently, we are trying to modify our HMM system for splicing acceptor site classification. At the same time, we are going to develop our Hidden Markov Models for exon, intron, start site and stop site recognition. Then we can integrate our models into an effective and accuracy system for gene structure detection.

6. References

[1] B. Alberts, D. Bray, J. Lewis, M. Raff, K. Roberts, and J.D. Watson, *Molecular biology of the cell*, 3rd ed. Garland Publishing, Inc. New York and London. 1994.

[2] S. Audic and J. Claverie, “Detection of eukaryotic promoters using Markov transition matrices”, *Computers and Chemistry*, 21:223-227, 1997.

[3] T.L. Bailey, M.E. Baker, and C. Elkan, “An artificial intelligence approach to motif discovery in protein sequences: Application to steroid dehydrogenases”, *J. Steroid Biochemistry*, 62(1):29-44, 1997.

[4] M. Borodovsky and J. McIninch, “GENMARK: Parallel gene recognition for both DNA strands”, *Computers and Chemistry*, 17:123-133, 1993.

[5] M. Burset and R. Guigo, “Evaluation of gene structure prediction programs”, *Genomics*, 34(3):353-367, 1996.

[6] R. Guigo, “Computational gene identification: An open problem”, *Computers and Chemistry*, 21(4):215-222, 1997.

[7] J. Henderson, S. Salzberg, and K.H. Fasman, “Finding Genes in DNA with a Hidden Markov Model”, *Journal of Computational Biology*, 4(2):127-141, 1997.

[8] A.V. Lukashin and M. Borodovsky, “GeneMark.hmm: New solutions for gene finding”, *Nuleic Acids Research*, 26(4):1107-115, 1998.

[9] S.L. Salzberg, “A method for identifying splice sites and translational start sites in eukaryotic mRNA”, *Computer Applications in the Biosciences*, 13(4):365, 1997.

[10] J. T. L. Wang, T. G. Marr, D. Shasha, B. A. Shapiro, G. W. Chirn, and T.Y. Lee, “Complementary classification approaches for protein sequences”, *Protein Engineering*, 9(5):381-386, 1996.

[11] J. T. L. Wang, S. Rozen, B. A. Shapiro, D. Shasha, Z. Wang, and M. Yin, “New Techniques for DNA Sequence Classification”, *Journal of Computational Biology*, 6(2):209-218, 1999.

[12] J. T. L. Wang, B. A. Shapiro, and D. Shasha, editors, *Pattern Discovery in Biomolecular Data: Tools, Techniques and Applications*, Oxford University Press, New York, New York, 1999.

[13] M. Yin and J. T. L. Wang, “Algorithms for splicing junction donor recognition in genomic DNA sequences”, *Proceedings of the IEEE International Joint Symposia on Intelligence and Systems*, pages 169-176, Rockville, Maryland, May 1998.