Automatic Gene Recognition without Using Training Data

Kiyoshi Asai [1] asai@etl.go.jp
Yutaka Ueno [1] ueno@etl.go.jp
Katunobu Itou [1] kito@etl.go.jp
Tetsushi Yada [2] yada@tokyo.jst.go.jp

[1] Electrotechnical Laboratories
1-1-4 Umezono, Tsukuba 305, Japan
[2] Japan Science and Technology Corporation
5-3 Yonbancho, Chiyoda-ku, Tokyo 102 Japan

Abstract

In this paper, we propose a new approach for gene recognition, which uses no training data *for the recognizer. In this approach, we start from a simple model, which only uses the knowledge of start codons and the stop codons, then the recognition of the DNA sequences by the recognizer and the training of the parameters of the recognizer by the result of the recognition are repeated. We applied this* parse and train *approach to the complete genome sequence of cyanobacterium, and achieved the almost same recognition rate with the case of using the whole sequence as training data. This results open the possibility to use automatic gene annotation system in the early stage of sequencing projects.*

1 Introduction

1.1 Gene Recognition and hidden Markov models

The sequencing projects are producing large sequence data during their progress. In order to understand the *meaning* of the sequences, it is necessary to develop effective computational systems to detect the genes in the DNA sequences. The exact locations of the genes and the splicing patterns are proved by experiments, but if computational gene finding systems can predict the genes correctly, time-consuming experiments may be reduced. There have been proposed a number of systems for finding genes. For example, GENMARK [4], FGENEH [26], GeneID [13], GeneParser [27], Genie [21], GRAIL [29], GeneHacker [31], EcoParse [19], HMMgene [20], SORFIND [16], GenLang [9], Morgan [24], VEIL [14], Procrustes [11], MZEF [32], GENSCAN [6].

Because genes have a structure like a language, computational linguistic methods are effective in order to understand their structure [9, 27]. However, the components and the rules of the *DNA language* behave as though non-deterministic, it is necessary to combine statistics and computational linguistics for the *parsing* of DNA. That is why hidden Markov models (HMM) are becoming widely used for gene recognition [6, 19, 20, 21, 30, 31]. In order to build a stochastic *DNA language* by using HMMs, we model the components of the gene structures by HMMs and connect them by the rules which represent the gene structures. Because HMMs have some limitations to express the positional correlations of the bases, the components are often made by the other methods, like artificial neural nets. The generalized HMMs allow such non-HMM models behave as a part of stochastic parsing [6, 21].

1.2 Previous Works

The authors have been building gene finding systems, **GeneHacker** (for prokaryote) and **GeneDecoder** (for eukaryote) using a parsing technique by a stochastic grammar based on hidden Markov models (HMMs). The main statistics between HMMs are codon bigrams, which is a first order Markov model (not *hidden* Markov model) of codons. The codon bigrams include the hexamer information in *reading frames*, but it is not exactly same to using simple hexamer or 5th order Markov model. The recognition accuracies by base counts were over 90% (cyanobacterium, [2, 31]) and over 80% (human, [3]). The authors have not implemented direct homology search of the protein database in their systems. Instead of using homology search by the protein database, the authors have implemented protein motif dictionary as a part of these systems [2, 3].

1.3 Building Recognizer without Using Annotated Data

All gene finding systems, including the previous works of the authors, have been using the annotations of the DNA sequences, which describe the partial or total structures of the genes, to decide the parameters of the systems. The important features of the components of the gene structures, such as codon usages, di-codon usages, GC contents and the signal patterns are extracted from these annotated data, and used as the parameters of the systems. The prediction methods vary depending on the systems, but all systems require *training data*, whose gene structures are annotated in advance, for the determination of the parameters of the gene finding systems. However, enough amount of training data would not often exist in the early stage of a sequencing project. If we can construct a gene recognizer without using annotated data, it would be very helpful for the determination of genes in sequencing projects.

In this paper, we propose a new approach, which uses *no training data* to build the gene finding system. In this approach, we perform the following *parse and train* process iteratively. We start with a very simple model, which only use what are the base-triplets of the start codons and of the stop codons. We predict the gene structures of DNA sequences by this simple model. We calculate the statistics of coding/non-coding regions using the recognition results. The parameters of the next stage model are decided using those statistics. We predict again the gene structures of DNA sequences by this new model. We calculate again the statistics of coding/non-coding regions using the new recognition results. The parameters of the next stage model are decided using those new statistics. We repeat these process until the parameters and the recognition results converge.

2 Data

We used the whole prokaryotic genome sequence (3,573,470 bases) of a unicellular cyanobacterium, *Synechocystis* sp. strain PCC6803 for the gene recognition. The sequence is divided into 27 entries in GenBank (D90899 - D90917, SYCCPNC, SYCSLLLH, SYCSLRB, SYCSLRF, SYCSLLE, SYCSLRA, SYCSLRD, SYCSLRG) with potential protein coding regions in the annotations ([15, 18]). A subset of the same data, contiguous 1M bases of cyanobacterium, has been analyzed by computer analyses by several researchers [2, 15, 31].

We used the sequence data of all 27 entries for our recognition test. The recognition results of the whole sequence are used to decide the parameters of the recognizer in each iteration. Therefore, the recognizer use the information of the whole sequence. However, the annotations in GenBank are used only to calculate the performance of the recognition results, because our method requires no advance annotations for training the recognizer. Because our recognizer parses both the direct strand and the complimentary strand at the same time, the annotations of the CDSs in both strands were used for validation.

Figure 1: Overview of the gene recognizer. *The genes on the direct strand and the ones on the complimentary strand are recognized by a single parse. The genes on the direct strand consist of the start codon, the internal codons and the stop codon in the normal order, but on the complimentary strand the genes are recognized as the series of the complimentary stop codon, complimentary internal codons and the complimentary start codon (reverse order).*

3 System

3.1 The Recognizer

The overview of our gene recognizer is shown in Figure 1. The system consists of direct strand model, complimentary strand model and the intergenic model. Each component in the diagram is an HMM, which matches to the subsequences of the given sequences in stochastic manner. The log-probabilities of the matches of the HMMs and the subsequences play the roles of the scores for the *parsing* based on dynamic programming.

The direct strand model is a combination of the start codons, the internal codons and the stop codons. Start codons and stop codons are simple 3-state HMMs. For example, the model of the start codon 'ATG' is a 3-state HMM, whose three states correspond to 'A', 'T' and 'G,' and always matches to the sequence 'ATG.' If we model the internal codons by the bigram of the codons, the bigram is a large HMM which has inter-connected codon HMMs (3-state) as its components. The transition probabilities between these codon HMMs in the codon bigram are the probabilities that each codon appears after the specified codon in the internal CDS sequences.

The components appears in reverse order in the complimentary strand model, because the gene recognizer reads the sequence of the direct strand. The complimentary strand model is a combination of the complimentary stop codons, the complimentary internal codons and the complimentary start codons in that order. In the complimentary strand model, the codons are expressed in 'reverse complimentary' manner. For example, the stop codons of the complimentary strand model are 'TTA', 'CTA' and 'TCA', which are the reverse compliments of 'TAA', 'TAG' and 'TGA.' The intergenic region model is also an HMM, which is a one-state-HMM in current implementation.

These transition probabilities in the direct/complimentary strand models and the output proba-

bilities of the intergenic region model are set to be *flat* for the initial recognizer. During the iteration of *parse and train* process, these probabilities are updated by the statistics of the prediction results.

The gene recognizer, which consists of component HMMs, itself is a stochastic model and produces the DNA sequence in a stochastic manner. The *parsing* of the target DNA sequence is the process to estimate the best sequence of transitions of hidden states, which outputs the target DNA sequence. We can annotate the target DNA sequence according to the estimated series of hidden states, which are classified as the start codon, the internal codon, etc. The coding regions on the direct strand and the ones on the complimentary strand are recognized by a single parse of the sequence of the direct strand, because the recognizer has both direct strand model and the complimentary strand model. The coding regions are assumed not to overlap each other in this model, but the edges of the coding regions sometimes overlap according to the annotation of the GenBank entries.

3.2 Parse and Train

The *parse and train* of the recognizer is an iterative process to improve the parameters of the recognizer by the recognition results produced by the recognizer with previous parameters. It requires no training data for the iteration, but use the information of the non-annotated DNA sequence by finding the gene structures of the sequence iteratively.

The *parse and train* proceeds as follows:

- The initial recognizer. The base-triplets of start/stop codons are given. The internal codon models in direct/complimentary strands are set to be *flat*. The probabilities for the codons to appear after the start codon are all set to be equal. If the codon models are bigrams, the transition probabilities between codons are all set to be equal. The output probabilities for intergenic model are set to be the base frequencies of the whole sequence.

- The first recognition. The whole sequence is parsed by the initial model using dynamic programming. The annotations of the coding regions are given as the result of the recognition.

- Update the parameters. Calculate the statistics and update the parameters of codon models and of the intergenic model. The probabilities for the codons to appear after the start codon are set to be the frequencies of these codons in the predicted coding regions. If the codon models are bigrams, the transition probabilities between codons are set to be the di-codon statistics of the predicted coding regions. The output probabilities for intergenic model are set to be the base frequencies in the predicted intergenic regions.

- Update the recognition. The whole sequence is parsed by the new model. The new annotations of the coding regions are given as the result of the recognition.

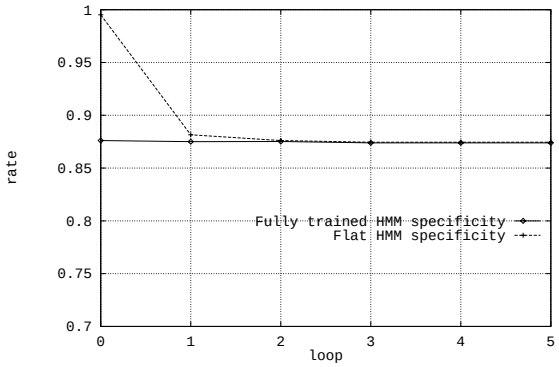
- Repeat previous two procedure until conversion.

Figure 2: Sketch of parse and train process *The following two processes are repeated. The recognizer predicts the coding regions and annotates the DNA sequence. The parameters of the recognizer are calculated from the annotation produced by the prediction.*

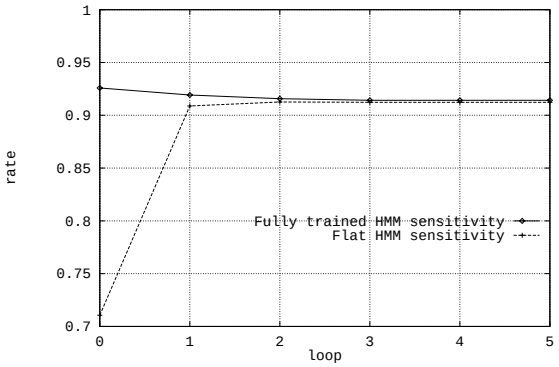
4 Results

Parse and train of the recognizer was tested by the whole sequence of the cyanobacterium data described in Section 2. First, we tested the recognizer with internal codon bigram model, starting with *flat* model as described in Section 3. In order to validate the *parse and train* method, we also tested the process beginning with the *fully trained* initial model, whose parameters are determined by using the GenBank annotations of the whole sequence. The plots of the recognition rates of base counts, sensitivities and specificities of CDSs during the repeats for *flat*/*fully trained* initial models are shown in Figure 3. The more precise statistics after five iterations for *flat* initial model are shown in Table 1. As shown in Figure 3, the recognition accuracies of *flat* initial model and *fully trained* initial model are almost same, and as high as 95% after three iterations. This results show the *parse and train* of the recognizer without training data succeeded, and the performance of the *parse and train* recognizer reached the upper limit of the given model.

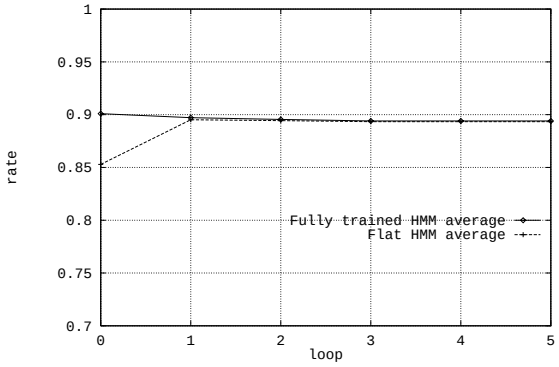
According to statistic analysis, the internal codon models should use di-codon statistics rather than single codon statistics [31]. However, single codon statistics can be better for *parse and train* method because we begin with very little information. We tested the same *parse and train* with single codon models. The results are shown in Figure 4. Although the converged recognition accuracies of *flat* and *fully trained* are also same, but the performance in CDS level is much worse than the case of the di-codon models. Because the recognition accuracies of single codon models are not good enough even if we begin with *fully trained* model, we clearly have to use di-codon models for the recognition of the genes of cyanobacterium with high performance.



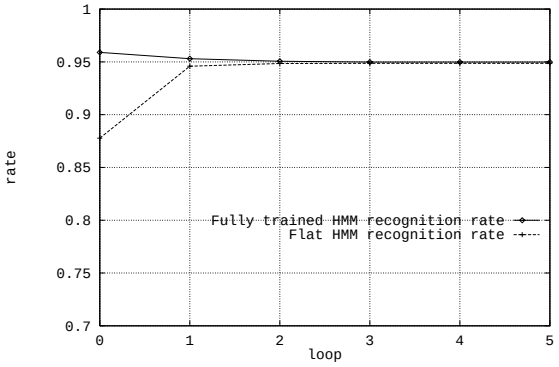
(a) Specificities of CDSs.



(b) Sensitivities of CDSs.

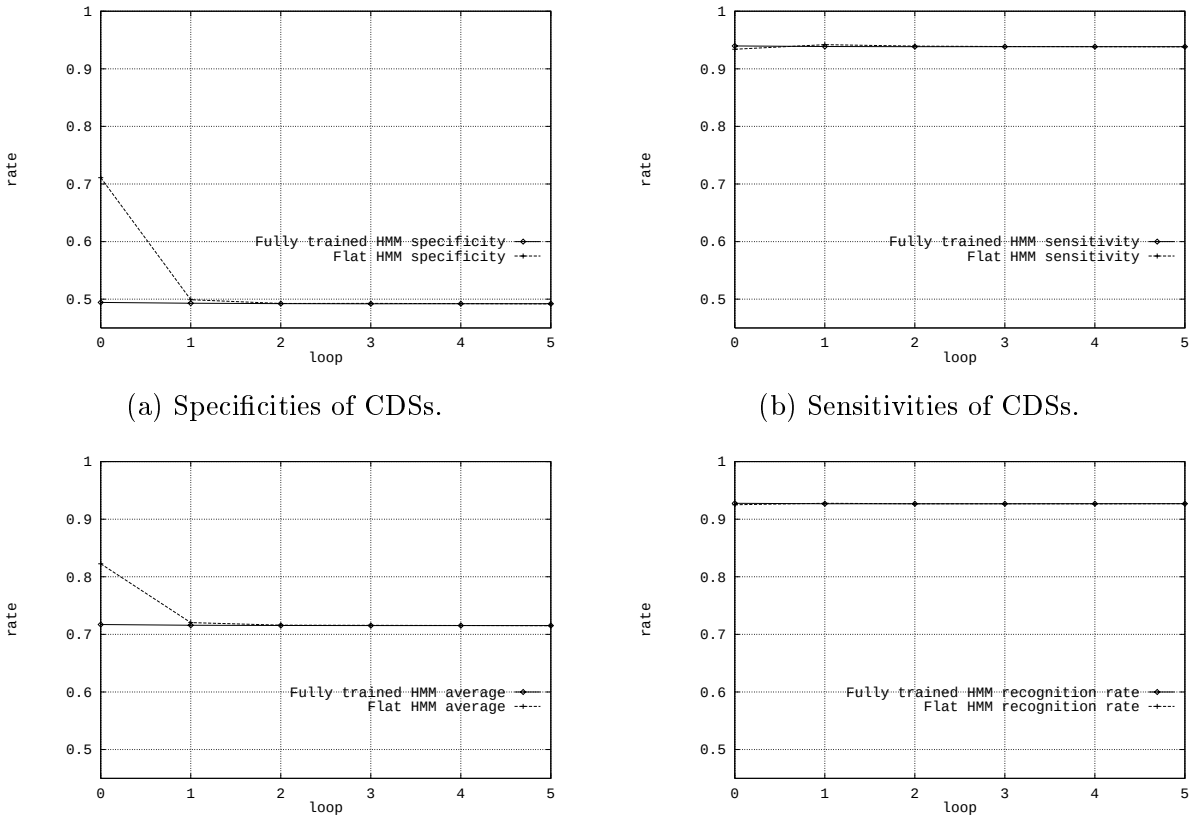


(c) Average of (a) and (b).



(d) Recognition rate in bases.

Figure 3: The recognition results of di-codon model. The recognition results of di-codon model. The predicted CDSs which have same stop codons with the annotate of GenBank are counted as CDS hits in CDS level. The sensitivities and the specificities are calculated in CDS level. The recognition rate in base level is the ratio of the sum of the number of correctly predicted CDS bases and the number of correctly predicted non-CDS bases divided by the total number of the entire bases. The *reading frames* were considered to determine the number of correctly predicted CDS bases.



(a) Specificities of CDSs.

(b) Sensitivities of CDSs.

(c) Average of (a) and (b)

(d) Recognition rate in bases.

Figure 4: The recognition results of single codon model.

Table 1: Recognition performance of *parse and train* model (no training data) after five iterations.

		strand		
		total	direct	complimentary
Base	Level			
	Sequence Length	3,573,470		
	True CDS	3,101,662	1,625,981	1,475,681
	Predicted CDS	3,004,811	1,568,097	1,436,714
	CDS hit	2,959,337	1,544,297	1,415,039
	Sensitivity	95.4%	95.0%	95.8%
	Specificity	98.5%	98.5%	98.5%
CDS	Level			
	True CDS	3,169	1,661	1,508
	Predicted CDS	3,306	1,713	1,593
	CDS hit	2,891	1,503	1,388
	Sensitivity	91.2%	90.5%	92.0%
	Specificity	87.5%	87.7%	87.1%

5 Discussions

Parse and train of HMMs are often necessary in speech recognition. In order to build accurate phoneme HMMs, we need large number of training data which have annotation of the boundaries of the phonemes. However, it is difficult to get a large number of annotated data. Therefore, a large number of non annotated (but usually the transcriptions are known) with a small number of annotated data are used for the construction of the phoneme HMMs.

In this paper, we tried a same kind of *parse and train* for a gene recognition system. The results of the gene recognizer for cyanobacterium show that we can predict CDSs around 95% without training data. The obvious but a difficult next step is to apply the *parse and train* approach to the eukaryotic DNA sequences. Because the structures of the eukaryotic genes are more complicated, it may be difficult to learn the parameters of the gene recognition system by *parse and train*. However, the idea of using the annotation produced by the recognition for the training of the parameters of the recognizer can be combined with the standard methods. We can begin the *parse and train* process with small number of training data, which itself is insufficient to construct a good recognizer. It should be particularly important for the early stage of sequencing projects.

6 Conclusions

We have proposed a new approach for gene recognition, which uses *no training data* for the recognizer based on hidden Markov models. It only uses the knowledge of start/stop codons for the initial model of the recognizer, and learns the parameters of the model by *parse and train*. We tested the approach to the whole genome sequence of cyanobacterium and achieved around 95% of recognition rate, which is only slightly worse than the case of using the whole sequence as the training data. This result implies that the proposed *parse and train* approach is effective for the recognition of prokaryote genes, and that automatic gene annotation is possible in the early stage of sequencing projects for prokaryotic sequences.

Acknowledgments

This work was supported in part by a Grant-in-Aid (08283101:"Genome Science") for Scientific Research on Priority Areas from The Ministry of Education, Science, Sports and Culture of Japan, and in part by Real World Computing (RWC) Program of The Ministry of International Trade and Industry of Japan. The authors thank Dr. Nobuyuki Otsu, Dr. Tetsuro Moriya and the members of Genome Informatics Group of Electrotechnical Laboratories for the support and the discussions.

References

[1] Asai,K.; Handa,K. and Hayamizu,S.: "Genetic Information Processing by Stochastic Model: HMM for Secondary Structure Prediction of Protein," *Genome Informatics,* **2**, 144-147 (in Japanese, 1991).

[2] Asai,K.; Yada,T. and Itou,K.: "Finding Genes by Hidden Markov Models with a Protein Motif Dictionary," *Genome Informatics,* **7**, 88-97 (1996).

[3] Asai,K.; Ueno,Y; Itou,K. and Yada,T.: "Recognition of Human Genes by Stochastic Parsing," *PSB98,* to appear.

[4] Borodovsky,M. and McIninch,J.: "GENMARK: parallel gene recognition for both DNA strands," *Comp. Chem.* **17**, 123-133 (1993).

[5] Bucher,P.: "Weight matrix descriptions of four eukaryotic RNA polymerase II Promoter elements derived from 502 unrelated promoter sequences," *J.Mol.Biol.* **202**, 563-578 (1990).

[6] Burge,C. and Karlin S.: "Prediction of Complete Gene Structures in Human Genomic DNA," *J.Mol.Biol.* **268**, 78-94 (1997).

[7] Burset,M and Guigó,R.: "Evaluation of gene structure prediction programs," *Genomics*, **34**, 353-367 (1996).

[8] Fickett,J.: "Assessment of protein coding measures," *Nucl. Acids Res.* **20**, 6441-6450 (1992).

[9] Dong,S. and Searls,D.B.: "Gene structure prediction by linguistic methods," *Genomics*, **23**, 540-551 (1994).

[10] Fujiwara,Y.; Asogawa,M. and Konagaya,A.: "Stochastic Motif Extraction Using Hidden Markov Model," *ISMB94*, **2**, 121-129 (1994).

[11] Gelfand,M.S.: Mironov,A.A. and Pevzner,P.: "Gene recognition via spliced alignment," *Proc. Natl. Acad. Sci. USA*, **93**, 3015-3019 (1996).

[12] GenBank. Genetic sequence data bank, release 92.0. *Technical report, BBN Laboratories, U.S.A.* (1995).

[13] Guigó,R.; Knudsen,S.; Drake,N. and Smith,T.: "Prediction of gene structure," *J. Mol. Biol.* **226**, 141-157 (1992).

[14] Henderson,J.; Salzberg,S. and Fasman,K.: "Finding Genes in DNA with a Hidden Markov Model," *J. Comp. Biol.* **4**(2), 127-141 (1997).

[15] Hirosawa,M.; Kaneko,T.; Tabata,S.; McIninch,J.D.; Hayes,W.S.; Borodovsky,M. and Isono,K.: "Computer survey for likely genes in the one megabase contiguous genomic sequence data of *Synechocystis* sp. strain PCC6803," *DNA Res.*, **2**, 239-246 (1995).

[16] Hutchinson,G.B. and Hayden, M.R.: "The prediction of exons through an analysis of spliceable open reading frames." *Nucleic Acids Research, 20:13* 3453-3462(1992).

[17] Itou,K.; Hayamizu,S. and Tanaka,H.: "Continuous Speech Recognition by Context-Dependent Phonetic HMM and an Efficient Algorithm for Finding N-best Sentence Hypotheses," *ICASSP-92*, I-21–24 (1992).

[18] Kaneko,T.; Tanaka,A.; Sato,S.; Kotani,H.; Suzuki,T.; Miyajima,N.; Sugiura,M. and Tabata,S.: "Sequence analysis of the genome of the unicellular cyanobacterium *Synechocystis* sp. strain PCC6803 sequence features in the 1Mb region from map positions 64% to 92% of the genome," *DNA Res.*, **2**, 153-166 (1995).

[19] Krogh,A.; Mian,I.S. and Haussler,D.: "A hidden Markov model that finds gene in E.coli DNA," *Nucleic Acids Res.*, **22**, 4768-4778 (1994).

[20] Krogh,A.: "Two methods for improving performance of an HMM and their application for gene finding," *ISMB97*, **5**, 179-186 (1997).

[21] Kulp,D.; Haussler,D.;Reese,M.G. and Eeckman,F.H.: "A Generalized Hidden Markov Model for the Recognition of Human Genes in DNA," *ISMB96*, **4**, 134-142, (1996).

[22] Kulp,D.; Haussler,D.;Reese,M.G. and Eeckman,F.H.: "Integrating Database Homology in a Probabilistic Gene Structure Model," *PSB97*, **2**, 232-244. (1996).

[23] Sakakibara,Y.; Brown,M.; Mian,I.S.; Underwood,R. and Haussler,D.: "Stochastic context-free grammars for modeling RNA," *Proceedings of 27th HICSS*, **V**, 284-293 (1994).

[24] Salzberg,S.; Delcher,A.; Fasman,K. and Henderson,J.: "A Decision Tree System for Finding Genes in DNA," *Technical Report 1997-03, Dep. Comp. Sci., Johns Hopkins Univ. (19 97)*

[25] Snyder,E.E. and Stormo,G.D.: "Identification of protein coding regions in genomic DNA sequences: An application of dynamic programming and neural networks," *Nucleic Acids Res.* **21**, 607-613 (1993).

[26] Solovyev,V.V.; Salamov,A.A. and Lawrence,C.B: "Predicting internal exons by oligonucleotide composition and discriminant analysis of splicable open reading frames," *Nucl. Acid. Res.* **22**, 5156-5163 (1994).

[27] Stormo,G.D. and Haussler,D.: "Optimally parsing a sequence into different classes based on multiple types of evidence," *ISMB94*, **2**, 47-55 (1994).

[28] Wu,T: "A segment-based dynamic programming algorithm for predicting gene structure," *J. Comp. Biol.* **3**(3), 375-394 (1996).

[29] Xu,Y.; Einstein, J.R.; Mural,R.J.; Shah,M. and Uberbacher,E.C.: "An improved system for exon recognition and gene modeling in human DNA sequence," *ISMB94*, **2**, 376-384 (1994).

[30] Yada,T. et al: "Extraction of Hidden Markov Model Representations of Signal Patterns in DNA Sequences," *PSB96*, **1**, 686-696 (1996).

[31] Yada,T. and Hirosawa,M.: "Gene Recognition in Cyanobacterium Genomic Sequence Data Using the Hidden Markov Model," *ISMB96*, **4**, 252-260 (1996).

[32] Zhang,M.Q.: "Identification of protein coding regions in the human genome by quadratic discriminant analysis," *Proc. Natl. Acad. Sci. USA* **94**, 565-568 (1997).