

Gene Identification using Phylogenetic Metrics with Conditional Random Fields

by

Ameya Nitin Deoras

Submitted to the Department of Electrical Engineering and Computer Science in
partial fulfillment of the requirements for the degree of

Master of Science in Electrical Engineering and Computer Science

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

May 2007

[June 2007]

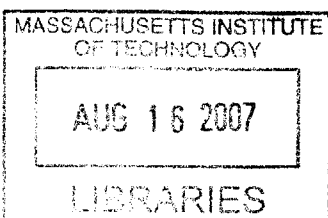
© Copyright 2007 Massachusetts Institute of Technology. All rights reserved.

The author hereby grants to M.I.T. permission to reproduce and distribute publicly
paper and electronic copies of this thesis and to grant others the right to do so.

Author _____
Department of Electrical Engineering and Computer Science
May 11, 2007

Certified by _____
Manolis Kellis
Assistant Professor, Distinguished Alumnus (1964) Career Development Chair
Thesis Supervisor

Accepted by _____
Arthur C. Smith
Chairman, Department Committee on Graduate Theses



BARKER



Room 14-0551
77 Massachusetts Avenue
Cambridge, MA 02139
Ph: 617.253.2800
Email: docs@mit.edu
<http://libraries.mit.edu/docs>

DISCLAIMER OF QUALITY

Due to the condition of the original material, there are unavoidable flaws in this reproduction. We have made every effort possible to provide you with the best copy available. If you are dissatisfied with this product and find it unusable, please contact Document Services as soon as possible.

Thank you.

The images contained in this document are of the best quality available.

Gene Identification using Phylogenetic Metrics with Conditional Random Fields

by

Ameya Nitin Deoras

Submitted to the Department of Electrical Engineering and Computer Science

May 2007

In partial fulfillment of the requirements for the degree of
Master of Science in Electrical Engineering and Computer Science

ABSTRACT

While the complete sequence of the human genome contains all the information necessary for encoding a complete human being, its interpretation remains a major challenge of modern biology. The first step to any genomic analysis is a comprehensive and accurate annotation of all genes encoded in the genome, providing the basis for understanding human variation, gene regulation, health and disease. Traditionally, the problem of computational gene prediction has been addressed using graphical probabilistic models of genomic sequence. While such models have been successful for small genomes with relatively simple gene structure, new methods are necessary for scaling these to the complete human genome, and for leveraging information across multiple mammalian species currently being sequenced. While generative models like hidden Markov models (HMMs) face the difficulty of modeling both coding and non-coding regions across a complete genome, discriminative models such as Conditional Random Fields (CRFs) have recently emerged, which focus specifically on the discrimination problem of gene identification, and can therefore be more powerful. One of the most attractive characteristics of these models is that their general framework also allows the incorporation of any number of independently derived feature functions (metrics), which can increase discriminatory power. While most of the work on CRFs for gene finding has been on model construction and training, there has not been much focus on the metrics used in such discriminatory frameworks. This is particularly important with the availability of rich comparative genome data, enabling the development of phylogenetic gene identification metrics which can maximally use alignments of a large number of genomes. In this work I address the question of gene identification using multiple related genomes. I first present novel comparative metrics for gene classification that show considerable improvement over existing work, and also scale well with an increase in the number of aligned genomes. Second, I describe a general methodology of extending pair-wise metrics to alignments of multiple genomes that incorporates the evolutionary phylogenetic relationship between informant species. Third, I evaluate various methods of combining metrics that exploit metric independence and result in superior classification. Finally, I incorporate the metrics into a Conditional Random Field gene model, to perform unrestricted *de novo* gene prediction on 12-species alignments of the *D. melanogaster* genome, and demonstrate accuracy rivaling that of state-of-the-art gene prediction systems.

Table of Contents

1	Introduction	6
2	Genes and Gene Models	8
2.1	Biological Signals	8
2.1.1	Genes and Proteins	8
2.1.2	Evolutionary Constraints	10
2.2	Graphical Models of Gene Structure	10
2.2.1	Hidden Markov Models.....	11
2.2.2	Bayesian Networks.....	13
2.2.3	Phylogenetic Hidden Markov Models.....	15
2.2.4	Conditional Random Fields	17
3	Discriminative Metrics for Classifying Protein-Coding Sequences	20
3.1	Existing Methods of Classifying Protein-Coding Sequences	20
3.1.1	Sequence-Based Methods	20
3.1.2	Evolutionary Signature-Based Methods	25
3.2	Novel Single-Sequence Metrics for Gene Identification.....	28
3.2.1	DiCodon Periodicity.....	28
3.2.2	Codon Composition Metric (CCM).....	29
3.3	Novel Discriminative Phylogenetic Metrics	30
3.3.1	Phylogenetic Codon Evolution	30
3.4	Integrated Metrics Incorporating Sequence and Evolutionary Signatures.....	35
4	Comparative Evaluation of Metrics on <i>de novo</i> Exon Classification.....	37
4.1	Protein-Coding Sequence Classification Data Set	38
4.2	Comparative Analysis of Single-Sequence Metrics	43
4.3	Comparative Analysis of Sequence-Based Alignment Metrics	46
4.4	Comparative Analysis of Evolutionary Metrics	50
5	Combinations of Metrics for Improved Classification	55
5.1	Posterior Probabilities Framework	56
5.2	Linear Discriminant Analysis	57
5.3	Support Vector Machines	58
5.4	Majority Voting	58

5.5	Combinatorial Posterior Combinations	59
6	<i>de novo</i> Gene Prediction with Conditional Random Fields	62
7	Future Work.....	66
8	Contributions	67
8.1	Novel Metrics for Exon Classification.....	67
8.2	Posterior Framework for Metric Combinations.....	68
8.3	Integrating Metrics into a Conditional Random Field Gene Model.....	68
9	References	69
	Appendix A:.....	73
	Appendix B:.....	74
	Appendix C:.....	78

Acknowledgments

To my wonderful fiancée, Laurie, for her unwavering faith and encouragement, and to my loving family for their support and guidance.

Thanks to Manolis, Matt, Mike and the whole CompBio lab for their advice and insights.

1 Introduction

One of the biggest accomplishments of the past decade has been the completion of the sequencing of the Human genome in 2003, thus making available the entire sequence of 3 billion base pairs that comprise our DNA [1]. While the sequence of the human genome contains all the information necessary to create a complete human being, its interpretation remains a major challenge of modern biology. The first step to any functional analysis of our genome is a comprehensive and accurate annotation of all genes encoded in the genome, providing the basis for understanding human variation, gene regulation, health and disease. However, our best estimates predict that genes occupy only 1.5% of the entire genome, making their discovery and annotation a challenging problem [2]. Furthermore, because of the high cost, and painstaking process, of manual annotation, there is an urgent need for computational solutions that can automatically and reliably annotate our genome. In addition, with the accelerating increase in the number of species currently being sequenced, there is very strong demand for computational systems that can reliably identify such functional regions by examining the evolution of related sequences.

In this thesis, I will address the problem of computational *de novo* Gene Prediction, which can be defined as the computational discovery and annotation of all the protein-coding genes present in a *target* genome given only the target genome sequence and alignments of genomes of related species called *informants*. Traditionally, the problem of computational gene prediction has been addressed using generative graphical models of genomic sequence. Most large scale systems such as N-SCAN [3] and EXONIPHY [4] have converged to a Generalized Hidden Markov model with a phylogenetic model of sequence evolution. However, while these systems have been moderately successful, their *de novo* gene prediction performance has not scaled as the number of aligned genomes has increased [3, 5]. Recently, *discriminative* models, such as Conditional Random Fields (CRFs) have emerged, which show

considerable promise over current *generative* models for gene prediction [6]. Being discriminative models, CRFs allow the incorporation of an arbitrary number of feature functions to evaluate different annotations of a sequence. However, while recent initial experiments with CRFs for gene prediction have mostly focused on model construction and the incorporation of non-probabilistic information (such as ESTs and homology matches) [5], there has not been much rigorous development of discriminative features for the *de novo* problem of gene prediction. In this work, I address this need for informed discriminative features (metrics) that can be incorporated into a CRF framework for superior gene prediction. I present a number of novel metrics that, by incorporating sequence biases as well as evolutionary signals, outperform existing metrics as well as scale with the number of aligned informant genomes. I evaluate the performance of these novel metrics and their combinations on the classification of gene sequences on the recently sequenced 12-species alignment of the Fruit Fly, the largest whole genome alignment data set for the animal kingdom [7]. Finally I incorporate these novel metrics into a CRF gene model and evaluate their performance on unrestricted *de novo* gene prediction.

The organization of this thesis is as follows. Chapter 2 provides the background material for understanding the gene prediction problem, as well as a review of graphical models used for gene prediction. In chapter 3, I summarize a number of existing discriminative methods for gene classification (metrics) and then present a number of novel discriminative metrics that incorporate sequence biases as well as evolutionary signatures. In chapter 4, I evaluate the performance of the proposed metrics on the classification of genes. In chapter 5, I describe a unified probabilistic framework for combining metrics and evaluate the performance of their combinations. In chapter 6, I build a complete CRF gene model informed by a single discriminative feature and evaluate its performance on the prediction of genes on unsegmented sequences. Finally, I propose improvements and extensions of these models in chapter 7 and summarize my contributions in chapter 8.

2 Genes and Gene Models

This chapter reviews the biology necessary to understand the gene prediction problem followed by a short history of the methods used to address this problem.

2.1 Biological Signals

The functional properties of genes impose a number of constraints on their sequences which can be used in their discovery in the genome. The following sections describe the two lines of evidence that can be used to discover these biases.

2.1.1 Genes and Proteins

From an information theoretic point of view, our DNA can be treated as a digital string of characters A, C, G, T, representing the four nucleotides. Functional elements or “cellular instructions” are coded within this string of characters. These instructions are recognized by cellular machinery and carried out during the growth and functioning of the cell. Genes are thought to comprise the largest group of functional elements in the DNA. Through the processes of transcription and translation the “instructions” in our genes are converted into proteins which then carry out most of the processes in the cell, including the expression and regulation of other genes. This process of transcription and translation of genes into proteins is described by the Central Dogma of biology as shown in Figure 2.1.

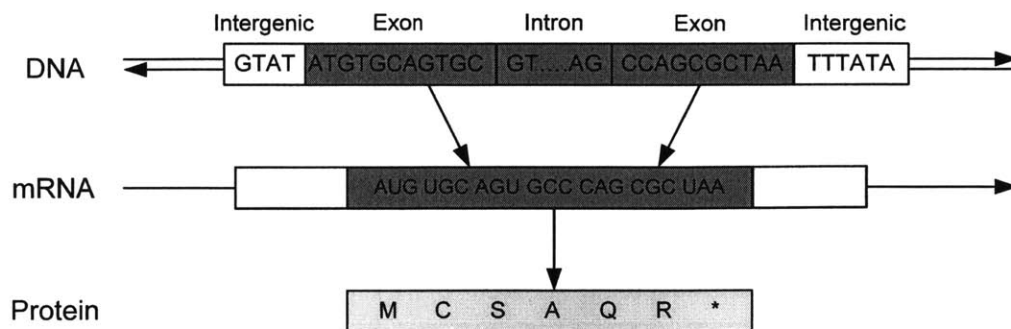


Figure 2.1: The Transcription of DNA into mRNA and translation into protein as described by the central dogma.

During transcription, the double stranded DNA is separated and an RNA template is generated by matching and chaining nucleotides complementary to that of DNA sequence. The introns are spliced out of the RNA chain to create a mature mRNA transcript. The mRNA nucleotides are then read in triplets (termed codons) and converted into a chain of amino acids to form proteins. The mapping between codons and amino acids is governed by the universal genetic code shown in Figure 2.2. Because nucleotides are read in triplets, the *reading frame* plays an important role in determining the translated protein product.

	T	C	A	G
T	TTT Phe (F) TTC " TTA Leu (L) TTG "	TCT Ser (S) TCC " TCA " TCG "	TAT Tyr (Y) TAC TAA Ter TAG Ter	TGT Cys (C) TGC TGA Ter TGG Trp (W)
C	CTT Leu (L) CTC " CTA " CTG "	CCT Pro (P) CCC " CCA " CCG "	CAT His (H) CAC " CAA Gln (Q) CAG "	CGT Arg (R) CGC " CGA " CGG "
A	ATT Ile (I) ATC " ATA " ATG Met (M)	ACT Thr (T) ACC " ACA " ACG "	AAT Asn (N) AAC " AAA Lys (K) AAG "	AGT Ser (S) AGC " AGA Arg (R) AGG "
G	GTT Val (V) GTC " GTA " GTG "	GCT Ala (A) GCC " GCA " GCG "	GAT Asp (D) GAC " GAA Glu (E) GAG "	GGT Gly (G) GGC " GGA " GGG "

Figure 2.2: The Universal Genetic Code

The genetic code is a redundant code in that more than one codon is translated into the same amino acid. Special codons such as ATG signal the ribosome to start the translation of the protein. Similarly at the end of the protein product, the stop codons TAA, TAG or TGA signal translation termination. The structure in the code that allows degeneracies in the third position of the codon, coupled with the preferred usage of certain codons in genes, creates patterns or biases in sequences that code for protein

(exons). These sequence-biases are exploited in both graphical models (Section 2.2) and discriminative metrics (Section 3) to discover protein-coding regions.

2.1.2 Evolutionary Constraints

During DNA replication, there is a very small but non-negligible error rate that leads to random insertions, deletions or substitutions in certain nucleotides in the DNA sequences. If these mutations occur so as to alter the function of a gene, they are usually detrimental to the survival of the cell. This creates an evolutionary pressure to preserve the functional aspect of DNA. Within genes, insertions or deletions that are not of length multiples of 3 are strongly selected against as they alter the reading frame resulting in a completely different protein product. Similarly, substitutions resulting in an in-frame stop codon, terminate the protein early, and are therefore also strongly selected against. However, mutations from one codon to another that encodes the same amino acid (a silent substitution) are preserved as they don't change the function of the protein. Because of these reasons, protein-coding regions of the genome face significantly different selective pressure than non-functional regions (where almost any type of mutation is equally likely). These biases in evolutionary patterns provide a wealth of information for detecting protein coding regions through the comparative analysis of the genomes of related species.

2.2 Graphical Models of Gene Structure

Historically, graphical modeling approaches have been the most accurate at annotating the genes within a genome. This section discusses the models used in state of the art systems as well as their advantages and disadvantages.

2.2.1 Hidden Markov Models

Hidden Markov Models have found widespread use in the computational biology community due to their versatility in modeling a variety of biological sequences as well as their simplicity of application. A hidden Markov model is a graphical model of sequential data that consists of a set of hidden states, a matrix of probabilities of transitions between states and a set of probabilities of emissions (outputs) for each state [8]. The model is said to *generate* a sequence of data by initially choosing a start state, emitting an output based on the emission probabilities for that state and then transitioning to the next state. The Markov property governs the state transitions, in that the probability of visiting a state at time (or position) t depends only on the state visited at time $t-1$. The emissions, in the case of DNA, are usually nucleotides, but could also be amino acids or codons. Given a sequence of emissions, the most likely state sequence that produced those emissions can be computed. When each state represents a genomic feature or label (such as *intron* or *exon*) the resultant state sequence also produces a parse or labeling of the input, thereby suggesting a sequence of classes that generated the outputs.

The decoding algorithm for a hidden Markov model computes the most likely state sequence given a sequence of emissions thought to have been generated by that model. If, as in the case of DNA models, each state represents a class, the maximum likelihood state sequence assigns a set of class labels to each emission, thereby producing a parse or annotation of the sequence. The major drawback of hidden Markov models is that, if the desired annotation is a segmentation of the emission sequence into segments belonging to certain classes, the length of those segmentations are forced to follow a geometric distribution, which may not match the length distribution of the segments. This motivates the use of generalized hidden Markov models or hidden semi-Markov models, which differ from HMMs in that they allow each state to produce a sequence of emissions of any length distribution. The process of decoding the state sequence now implies a segmentation of the sequence instead of a labeling, albeit at the increased cost of computation [9].

One of the most successful single sequence gene prediction systems, GENSCAN [10], is based on a generalized hidden Markov model of sequential DNA structure. The state transition diagram of GENSCAN is shown in Figure 2.3. The model contains one 'Intergenic' state and two sets of 'Gene' states for each DNA strand. The model is usually trained on annotated sequences from either the target organism or organisms closely related to the target organism. New sequences are then annotated by finding the maximum likelihood sequence of states through the model, and assigning each segment of the sequence the label of the state corresponding to that segment.

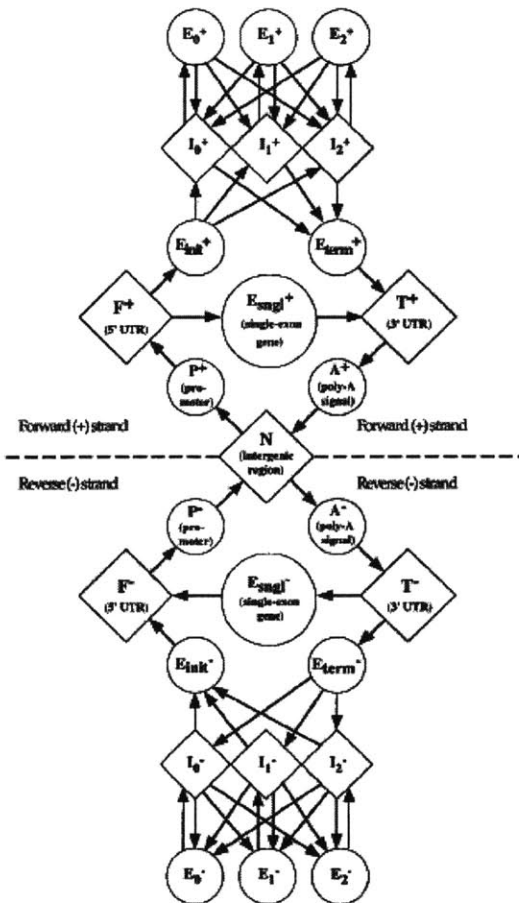


Figure 2.3: State Diagram of the Hidden Markov Model of GENSCAN

2.2.2 Bayesian Networks

In a hidden Markov model, the emission at a certain position (or time) is defined to depend only on the hidden state random variable at that position (or time). Such a conditional relationship is a small subset of a larger general class of graphical models called Bayesian networks [11].

A Bayesian Network is a graphical representation of conditional dependencies between a set of random variables. An example of a Bayesian network is shown in Figure 2.4. Each edge from one variable to another in the model encodes a conditional dependence of the child variable upon the parent. Two variables are considered conditionally independent given the value of a common parent. The conditional probability distribution of each random variable is given by a Conditional Probability Table (CPT) in the case of discrete random variables.

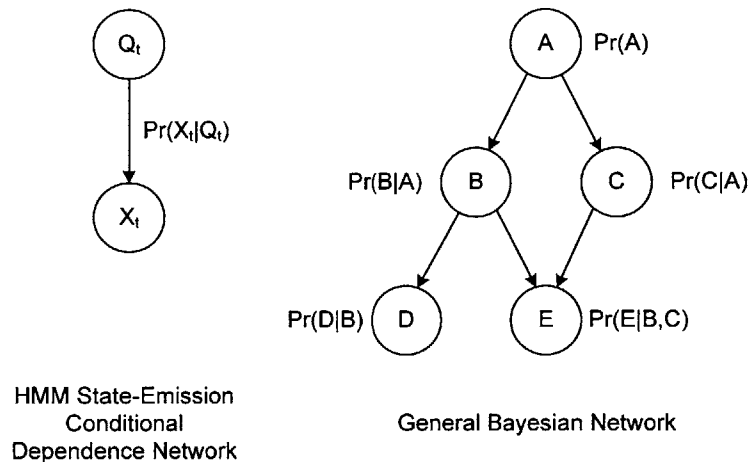


Figure 2.4: A Bayesian network representation of an HMM emission (left). An example of a Bayesian network encoding the conditional dependencies between a set of random variables A through E (right).

The advantage of a Bayesian network representation is seen when computing the joint likelihood of one or more variables in the network. Due to the conditional independence, the joint probability can be factored into a product of the CPTs encoded in the network. For example, for the network in Figure 2.4,

$$\Pr(ABCDE) = \Pr(A) \Pr(B|A) \Pr(C|A) \Pr(D|B) \Pr(E|B, C)$$

The marginal probability or likelihood of a subset of variables can also be computed given observed values of other variables in the network. Efficient families of algorithms exist that can compute the required marginal probabilities for any class of variables [11].

Bayesian networks play an important role in this discussion as they provide a convenient graphical model for representing evolution [12]. In an evolutionary lineage, every descendent depends only on its immediate ancestor and is conditionally independent of other descendants given a common ancestor. A Bayesian network is highly suited for representing such a relationship. In such networks, each node has only one parent, with the leaves representing modern day species and the parents representing ancestral species. If the nodes represent genetic sequences (which can be treated as discrete random variables), the conditional probability tables along each branch represent a model of sequence evolution along that branch of the phylogenetic tree. Figure 2.5 shows a phylogenetic tree relating Human, Chimp, Mouse and Dog and its Bayesian network representation.

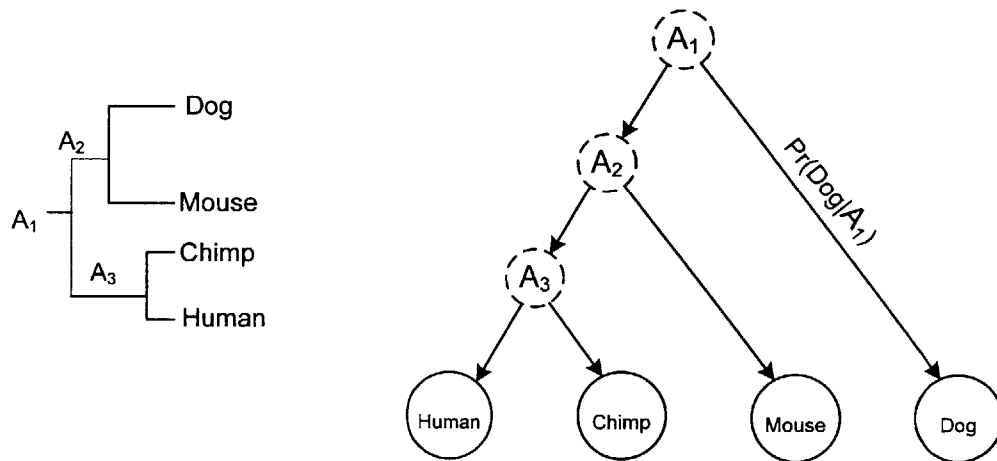


Figure 2.5: A phylogenetic tree relating Human, Chimp, Mouse and Dog (left) and its Bayesian network representation (right).

If the topology of the network is known, and the variables are observed, training the Bayesian network is reduced to estimating the conditional probability distributions that maximize an objective function (such as Maximum likelihood). If some of the variables are not observed, the parameters of the CPTs can

be approximately estimated by Expectation-Maximization. However, for phylogenetic Bayesian networks, a strong prior can be applied to the CPTs to obviate the approximate learning algorithm. Since each branch represents an evolutionary process, a mathematical model of evolution (eg. A Kimura model [13]) can be used to estimate each CPT by estimating the parameters of the model to fit that branch length and observed sequence. The resultant Bayesian network, usually termed a Phylogenetic Bayesian network, is what is commonly used in multiple-species gene predictors, as described in the next section.

2.2.3 Phylogenetic Hidden Markov Models

With the increasing availability of sequenced genomes of related species, gene prediction systems with an extended class of probabilistic models of generating alignments of sequence were found to significantly outperform single sequence models. Gene predictors such as EXONIPHY [4], TWINSCAN [14] and N-SCAN [3] were graphical models that augmented a hidden Markov model of genetic structure with a Bayesian network of nucleotide evolution at every sequence position [15]. The emissions in these models are columns of alignments of nucleotides (in the case of EXONIPHY and N-SCAN) or pairs of nucleotides (in the case of TWINSCAN). Such systems use a nucleotide model of evolution to train phylogenetic Bayesian networks of nucleotide evolution under different states of the model. The overall state space of the models remains relatively consistent with Figure 2.3.

Most of the state of the art gene predictors in current use are fundamentally based on some form of generative graphical model of phylogeny and sequence structure. Figure 2.6 summarizes the performance of these gene predictors on gene predictions in the Human and Fly genomes.

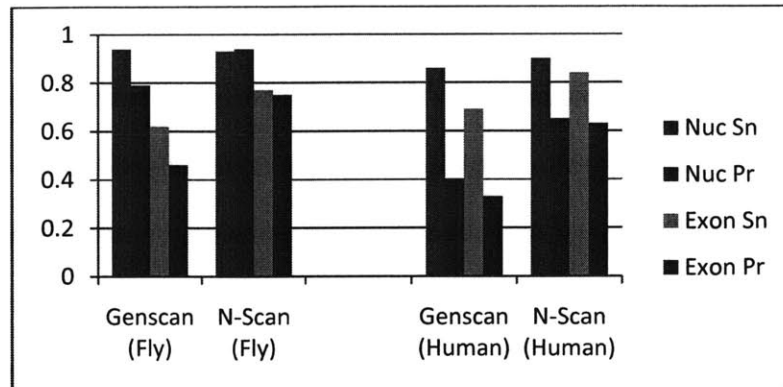


Figure 2.6 The *de novo* gene prediction performance of Genscan (single sequence) and N-Scan (alignment) gene models.

Despite their sophistication, these models suffer from some common drawbacks. For example, it has been documented that the performance of N-SCAN does not appreciably scale with an increase in the number of informants (aligned genomes) [3]. More serious, however, is that fundamentally, these models, attempt to model all characteristics of the underlying genome sequence and alignment. The result is that the systems model the probability of *generating* the sequence alignments as well as the annotation of the sequence. Namely, the models are trained to maximize the joint probability of the sequence and its annotations. However, during annotation of a new sequence, the desired annotation is one that maximizes the *conditional* probability of the annotation *given* the sequence.

$$\widehat{Annotation} = \operatorname{argmax}(\operatorname{Pr}(Annotation|Sequence)) = \frac{\operatorname{Pr}(Sequence, Annotation)}{\operatorname{Pr}(Sequence)}$$

Therefore, generative models have to trade off the likelihood of the sequence with that off the annotation given the sequence to maximize the joint probability. This often leads to suboptimal performance. Due to this mismatch between training and testing objectives, there has been a lot of recent interest in a different class of models, termed *discriminative models* that are trained to maximize only the conditional probability in the above equation. The following section discusses one particular class of such models, Conditional Random Fields.

2.2.4 Conditional Random Fields

Conditional Random Fields (CRFs) are discriminative graphical models of sequential data. Linear Chain Semi-Markov Conditional Random Fields (semi-CRFs) have been shown to be equivalent to Generalized Hidden Markov Models when the parameters are trained to maximize the joint probability of the annotation given the genomic sequence [6]. However, due to their discriminative structure, they can afford significant flexibility.

Being discriminative models, semi-CRFs are a graphical representation of the conditional probability $\Pr(\mathbf{S}|\mathbf{X})$ for a sequence \mathbf{X} with a corresponding segmentation (or annotation) \mathbf{S} [16]. \mathbf{S} is defined to be a set of triples $\{s_j\} = \{u_j, t_j, y_j\}$ where u , t and y represent the start position, end position and label (annotation) of a segment s . The conditional probability is defined in terms of feature functions, $g_k(s_j)$, which, under the semi-Markov model, depend only on a segment, its label and the label of the previous segment. Since the model does not generate outputs or emissions, the Markovian nature does not affect the observed input. Therefore the feature function can observe the entire genome sequence at any point. The conditional likelihood of the model is then given by,

$$\Pr(\mathbf{s} | \mathbf{x}, \Lambda) = \frac{1}{Z(\mathbf{x})} \exp \left(\sum_{k=1}^K w_k \left(\sum_{j=1}^{|\mathbf{s}|} g_k(y_j, y_{j-1}, t_j, u_j, \mathbf{x}) \right) \right)$$

where the weights w_k are used to weight the values of the feature functions and the function $Z(\mathbf{x})$ is a normalization function needed to make the result a valid probability. The schematic operation of a Linear Chain Semi-CRF on the annotation of an alignment of sequences is shown in Figure 2.7. The CRF proposes a number of possible segmentations of the input sequence and uses the feature functions to score the proposed segments. The overall score of a segmentation (or parse) is then the weighted sum of the scores of each of the feature functions on every segment. With the semi-Markov assumption and

the added restriction of a maximum segment length, the segmentation can be computed using dynamic programming by using the semi-Markov analogue of the Viterbi algorithm [17].

$$\hat{\mathbf{s}} = \arg \max_s \Pr(\mathbf{s} | \mathbf{x}, \Lambda) = \arg \max_{\{u,t,y\}} \left(\sum_{k=1}^K w_k \left(\sum_{j=1}^{|\mathbf{s}|} g_k(y_j, y_{j-1}, t_j, u_j, \mathbf{x}) \right) \right)$$

The key insight into the segmentation procedure is that the best segmentation of a sequence up to position i , is the concatenation of the best segmentation of the sequence up to position $i-d$ and the segment from $i-d$ to i , maximized over all possible values of d . The score of the best segmentation up to position i ending with label y is computed by,

$$V(i, y) = \begin{cases} \max_{y', d=1..L} V(i-d, y') + \sum_{k=1}^K w_k \left(\sum_{j=1}^{|\mathbf{s}|} g_k(y_j, y_{j-1}, t_j, u_j, \mathbf{x}) \right) & \text{if } i > 0 \\ 0 & \text{if } i = 0 \\ -\infty & \text{if } i < 0 \end{cases}$$

The segmentation can then be found by tracing back through the values of V that were maximized at each iteration.

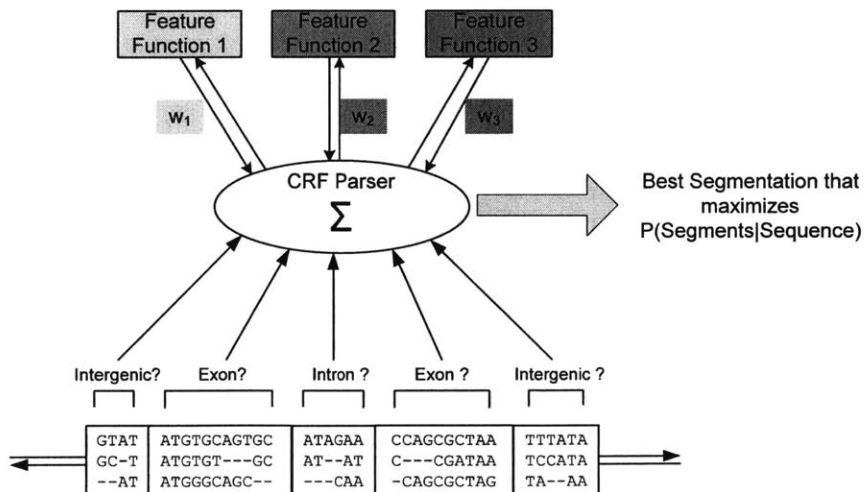


Figure 2.7: A schematic of the operation of a Semi-Markov Linear Chain CRF for the discriminative annotation of an alignment of sequences.

In addition to being trained to maximize the conditional annotation probability, CRFs offer many advantages over their generative counterparts for the gene prediction problem. Firstly, because the feature functions can examine the entire sequence at any point, the number of states in the gene model can be reduced, and incorporated directly into the feature functions. Secondly, the CRF has far fewer parameters than its generative counterparts, thereby requiring less data to train the weights of the model (assuming feature functions have already been trained) [18, 19]. Lastly the CRF offers the flexibility of including any number of feature functions, which can be nearly impossible with generative models.

Given the strengths of the model, the focus can now be shifted to the feature functions that provide the bulk of the discrimination in the CRF. The ideal candidate for a feature in a gene prediction CRF model must be very accurate at discriminating between protein-coding and non-coding sequences. An informed feature must therefore exploit as many patterns and biases in the coding-sequences that discriminate them from non-coding sequences. Ideally, the features should also incorporate phylogenetic information between the genomes comprising the alignment in such a way as to improve discrimination as the number of informant genomes is increased.

In the following chapter, we describe a number of methods (metrics) that have traditionally been used to discriminate between coding and non-coding sequences, which possess some of the characteristics that make them suitable candidates for feature functions in a semi-CRF gene predictor.

3 Discriminative Metrics for Classifying Protein-Coding Sequences

The biases in the nucleotide sequences in protein coding regions stem from two processes. The process of gene expression as well as the structure of the genetic code creates a *sequence bias* that favors the use of either certain triplets or certain nucleotides in certain positions. The other type of bias comes from selective pressure from the highly constrained process of *evolution* of protein coding sequences. Almost all methods of identifying or classifying protein coding regions from the sequence alone (methods that do not use homology or protein-database queries) exploit one or both of these biases.

The first section describes existing methods for classifying protein coding regions. In section 3.2, we propose novel single-sequence discriminative metrics for gene identification. In section 3.3 we describe novel metrics that discriminate between alignments of protein-coding and non-coding sequences based on their patterns of evolution. Finally, section 3.5 describes a metric that incorporates both sequence as well as evolutionary biases.

3.1 Existing Methods of Classifying Protein-Coding Sequences

3.1.1 Sequence-Based Methods

The bias in coding regions of the genome affects the DNA sequence in two ways. Firstly, due to the triplet nature of the genetic code, different statistical properties can be seen in each of the three different reading frames, resulting in a 3-periodic signal. Methods that exploit this signal will be said to use *inter-frame analysis*. The second characteristic is the composition of nucleotide triplets in the sequence. Methods that use this feature will be said to use *compositional analysis*.

i) 3-Base Periodicity and the FFT Metric

The FFT metric is a measure of the inherent 3-base periodicity present in a sequence of DNA. The source of this periodicity is a bias in the genetic code that favors certain nucleotides in certain positions (irrespective of the sequence of amino acids). The periodicity of each nucleotide character, is computed as the 1/3 frequency of the magnitude discrete Fourier transform (DFT) of the indicator sequence of that nucleotide character [20-22]. The indicator sequence is a binary sequence indicating the presence of a character at that position. The aggregate score for the sequence is then computed by summing the periodicity scores of the four nucleotide characters.

$$S[k] = \frac{1}{N} \left(|U_A[k]|^2 + |U_T[k]|^2 + |U_C[k]|^2 + |U_G[k]|^2 \right)$$

where $U_i[k]$ is the DFT of the indicator sequence for nucleotide i , N is the length of the sequence and k is the frequency at which the DFT is evaluated. Figure 3.1 shows the magnitude discrete Fourier transform of a protein coding and non-coding stretch of DNA from the common Fruit Fly. The sharp peak at 1/3 frequency reflects the inherent 3-base periodicity present in the coding sequence. Because it is purely an inter-frame metric, the FFT score is not affected by the order of amino acids in the sequence. Also, because it depends only on the magnitude of the DFT, the metric is agnostic about the reading frame of the sequence.

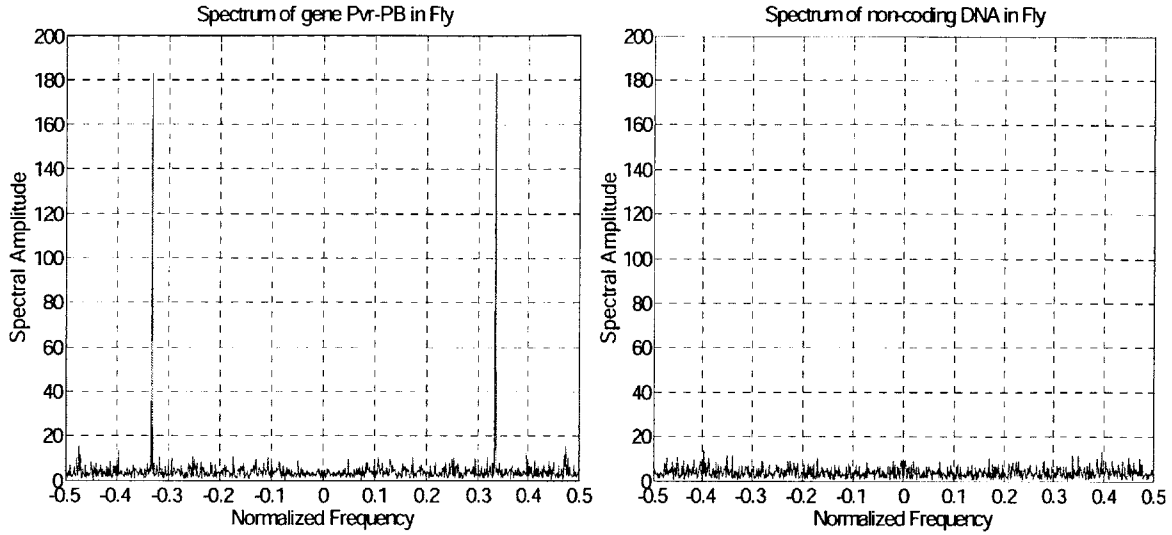


Figure 3.1: A sharp peak at 1/3 frequency in the Discrete Fourier Transform of coding-sequences versus non-coding sequences.

ii) Z Curve

Similar to the FFT score, the Z curve [23] is an inter-frame periodicity metric. It has been shown that the Z-curve method outperforms other single-sequence methods, such as Markov chains, codon usage, hexamer usage and the FFT on classifying short exons from Human genes [23, 24].

Given a DNA sequence, the Z curve computes a high dimensional vector comprised of frame specific mono-, di-, and tri-nucleotide occurrence. The Z curve score is then calculated as a projection of that vector onto a linear discriminant vector. Therefore, unlike the FFT, the Z curve must perform linear discriminant analysis on a training set of annotated protein coding and non-protein coding segments of DNA from the target species. The sequence statistics that comprise the Z curve are computed as follows.

Mono-Nucleotide Frequencies (9 parameters)

$$\begin{cases} x_i = (a_i + g_i) - (c_i + t_i) \\ y_i = (a_i + c_i) - (g_i + t_i) \\ z_i = (a_i + t_i) - (g_i + c_i) \end{cases} \text{ where } a_i \text{ is the frequency of adenine in reading frame } i. \\ x_i, y_i, z_i \in [-1, 1], \quad i = 1, 2, 3$$

Di-Nucleotide Frequencies (24 parameters)

$$\begin{cases} x_X = (p(XA) + p(XG)) - (p(XC) + p(XT)) \\ y_X = (p(XA) + p(XC)) - (p(XG) + p(XT)) \\ z_X = (p(XA) + p(XT)) - (p(XG) + p(XC)) \end{cases} \text{ where } p(CG) \text{ is the in-frame frequency of CG} \\ X = A, C, G, T$$

Tri-Nucleotide (Codon) Frequencies (36 parameters)

$$\begin{cases} x_{XY} = (p(XYA) + p(XYG)) - (p(XYC) + p(XYT)) \\ y_{XY} = (p(XYA) + p(XYC)) - (p(XYG) + p(XYT)) \\ z_{XY} = (p(XYA) + p(XYT)) - (p(XYG) + p(XYC)) \end{cases} \\ XY = A, C, G, T$$

Because of the linear discriminant analysis procedure, the parameters of the Z curve (the coefficients of the linear discriminant vector) can be “tuned” to the data set. The discriminant vector is also trained discriminatively, chosen to maximize the linear discrimination of the protein coding and non-coding sequences in the dataset. Therefore, the Z curve typically outperforms the FFT metric. The disadvantage of the Z curve is that its properties do not generalize well to alignments of sequences, as will be shown in Chapter 4.

iii) Interpolated Markov Models (IMM)

Interpolated Markov Models are a set of graphical models that are comprised of a number of Hidden Markov Models of varying order [25]. An n^{th} order Markov Model has state transitions that are governed by the previous $n-1$ states visited. Interpolated Markov Models actively weight the probabilities chosen by each of the component HMMs depending on the amount of training data available and the nature of the sequence being modeled [26]. IMMs were first introduced in prokaryotic gene prediction, where fixed order Markov models were found to be prone to overfitting to the training data in certain situations. In the context of protein-coding sequence classification, the generative IMMs are used to build a discriminative metric as follows. Using the GLIMMER package [26], an IMM Λ_P is trained on protein coding sequences and a model Λ_N on non-coding sequences. A new segment is scored by computing its likelihood of being generated by both models and taking the log ratio.

$$IMM(S) = \log \frac{\Pr(S|\Lambda_P)}{\Pr(S|\Lambda_N)}$$

The result is a discriminative measure of the posterior probability of the sequence being protein-coding.

iv) Codon Markov Chain

The codon Markov chain is a single-species model of first order codon sequence. By creating two such models for coding and non-coding sequences, the model captures the unequal patterns of occurrence of codon sequences in coding and non-coding regions. The Codon Markov Chain Metric (CMC) score for a sequence of DNA $S = \{S_1, S_2, \dots, S_n\}$, interpreted as a sequence of codons in the correct reading frame, is a log-likelihood ratio of the sequence being generated by each of the models.

$$CMC(S) = \log \frac{\Pr(S|CMC_C)}{\Pr(S|CMC_N)} = \log \frac{U_C(S_1) \prod_{j=2}^n B_C(S_j|S_{j-1})}{U_N(S_1) \prod_{j=2}^n B_N(S_j|S_{j-1})}$$

where U is a unigram probability of codon occurrence while B is a conditional codon-transition matrix of probabilities for each model. The values of each of the models are calculated simply by observing the occurrence of codons and pairs of codons in the training set for both the coding and the non-coding model.

Despite its simple structure the CMC score is a fairly accurate metric for classifying protein coding regions. It is also a discriminative metric since its score is the ratio of likelihoods of the sequence given the two models. CMC also plays an important role in combinations with other metrics. Because it is a single sequence metric it often contains information complimentary to purely comparative (evolutionary signature-based) metrics, especially in exons that are either not well conserved or misaligned.

3.1.2 Evolutionary Signature-Based Methods

In contrast to the single sequence methods, evolutionary methods predominantly exploit biases in the pattern of conservation in the alignments of sequence that indicate either coding or non-coding evolutionary constraints.

i) Sequence Conservation

Evolutionary constraints on functionally important regions of the genome (such as coding-regions) strongly select against mutations that result in a loss of function. Such evolutionary constraints do not usually apply to the majority of non-functional non-coding sequences. It has been observed that around 5% of the human genome has been conserved over 200 million years of evolution between mammals [2]. However, genes only comprise 1.5% of the human genome, suggesting that the majority of conserved sequence is functional but non-coding. However, sequence conservation can help discriminate between a large fraction of coding-sequences and non-coding sequences making it a good baseline to compare other metrics. For every column in the alignment of a segment, the conservation score for that column is calculated as the largest fraction of species with the same nucleotide. The conservation score for an entire alignment is obtained by averaging the conservation scores of each column.

ii) Reading Frame Conservation (RFC)

Reading Frame Conservation [27] is a measure of the degree to which the alignment of a segment captures the pressure to preserve the correct reading frame during evolution. It has been shown that the RFC measure alone can be used to accurately annotate the Baker's Yeast (*Sachharomyces Cerevisiae*) genome [27].

RFC examines the gap patterns in the alignment of an informant species sequence to the target sequence to determine the ratio of the number of nucleotides in incorrect frames to the number of

nucleotides in the correct reading frame. It essentially penalizes gaps that are not multiples of 3 by the number of nucleotides following the gap before a compensating gap length is seen. The RFC score for an alignment of a pair of sequences is simply the ratio of the number of in-frame nucleotides to the length of the sequence. In alignments of multiple informants, the RFC score for each pair-wise alignment to the target species' sequence is calculated. The resulting RFC score for the alignment is reported as the number of informants that have an RFC score of over .8 (a suggested cutoff for coding-sequences) [27].

iii) Codon Substitution Metric

The codon substitution metric, developed by Mike Lin [28], is a pair-wise metric that uses a model of probabilistic codon substitution in coding and non-coding regions to evaluate the observed substitutions in codons between a *target* species and an *informant* species. The models are trained on empirical codon substitution statistics gathered from regions of the genome known to be either protein-coding or non-coding. The resultant model is a 64x64 Codon Substitution Matrix [28], that encodes the probabilities,

$$CSM_C(T, I) = \log \Pr (I|T, I \neq T, Coding)$$

where T is a codon in the target sequence and I is a substituted codon in the informant sequence. I will use the subscript C to denote the coding-CSM and N to denote the non-coding CSM. An example of a CSM in protein-coding regions as well as in non-coding regions is shown in Figure 3.2. Notice that the coding model does not penalize silent substitutions between codons that code for the same amino acid. Also notice that stop codons are much more strongly conserved in the coding model than in the non-coding model. The CSM also captures the likelihood of substitution between codons that result in different amino acids that may share the same functional properties, similar to a BLOSUM matrix of amino acid substitutions [29].

The Codon Substitution Metric score for a pair-wise alignment of is then computed by summing the log likelihood ratios of all the non-conserved codon substitutions scored by the coding CSM and the non-coding CSM [28].

$$CSM_{PW}(\{T\}, \{I\}) = \sum_{i=1}^n CSM_C(T_i, I_i) - CSM_N(T_i, I_i) = \sum_{i=1}^n \log \frac{\Pr(I_i|T_i, I_i \neq T_i, Coding)}{\Pr(I_i|T_i, I_i \neq T_i, Non - Coding)}$$

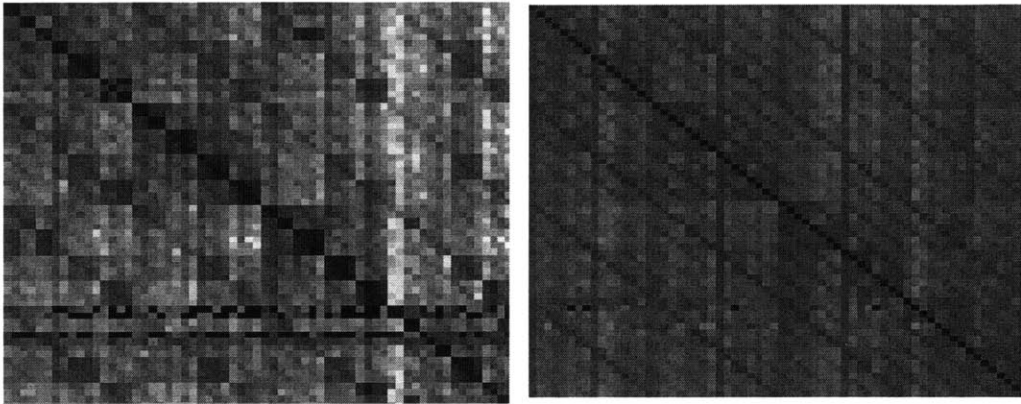


Figure 3.2: A CSM from protein-coding regions (left) and a CSM from non-coding regions (right).

3.2 Novel Single-Sequence Metrics for Gene Identification

In this section I present novel methods of discriminating between coding and non-coding sequences by exploiting the sequence bias in coding regions.

3.2.1 DiCodon Periodicity

The FFT metric from Section 3.1.1 can be shown to be an instance of a class of metrics that exploit inter-frame 3-base periodicity signals inherent in coding regions due to the triplet nature of the genetic code and the preferred use of certain codons. As the following derivation shows, an inter-frame periodicity computation can be applied to any frame-specific numeric quantity.

The discrete Fourier transform of a sequence $x(k)$ evaluated at $1/3$ frequency (in radians) is given by,

$$X\left(\frac{2\pi}{3}\right) = \sum_{k=0}^{n-1} x(k) e^{-j\frac{2\pi}{3}k}$$

Since $e^{-j\frac{2\pi}{3}k}$ is a periodic signal with period 3, $X\left(\frac{2\pi}{3}\right)$ can be rewritten as,

$$X\left(\frac{2\pi}{3}\right) = \sum_{k=0}^{\lfloor \frac{n}{3} \rfloor} x(3k) + \sum_{k=0}^{\lfloor \frac{n-1}{3} \rfloor} x(3k+1)e^{-j\frac{2\pi}{3}k} + \sum_{k=0}^{\lfloor \frac{n-2}{3} \rfloor} x(3k+2)e^{-j\frac{4\pi}{3}k} = x_1 + \alpha x_2 + \beta x_3$$

where $\alpha = \exp(-j*2\pi/3)$, $\beta = \exp(-j*4\pi/3)$ and X_i is the sum of the values of $x(k)$ in the i^{th} frame.

The magnitude DFT at $1/3$ frequency is therefore given by,

$$\left|X\left(\frac{2\pi}{3}\right)\right|^2 = X\left(\frac{2\pi}{3}\right) \bar{X}\left(\frac{2\pi}{3}\right) = (x_1 + \alpha x_2 + \beta x_3)(x_1 + \bar{\alpha} x_2 + \bar{\beta} x_3)$$

$$\left|X\left(\frac{2\pi}{3}\right)\right|^2 = x_1^2 + |\alpha|^2 x_2^2 + |\beta|^2 x_3^2 + (\alpha + \bar{\alpha}) x_1 x_2 + (\alpha \bar{\beta} + \beta \bar{\alpha}) x_2 x_3 + (\beta + \bar{\beta}) x_3 x_1$$

$$\left|X\left(\frac{2\pi}{3}\right)\right|^2 = x_1^2 + x_2^2 + x_3^2 - x_1 x_2 - x_2 x_3 - x_3 x_1 \quad (\text{Equation 3.1})$$

which is a general formula for the computation of 3-base periodicity of any sequence. In the case of the FFT metric, for each nucleotide α , x_i is simply the normalized number of occurrences of nucleotide α at

frame position i . However, such a simple numerical transformation may not capture all of the discriminative bias in the genetic code. A better discriminative transformation could be provided by a discriminative metric that was sensitive to the correct reading frame for coding regions but not for non-coding regions. The DiCodon Periodicity Metric is a measure of the inter-frame periodicity in the log-likelihood ratios of a modified Codon Markov Chain (CMC) model described in Section 3.1.1. The CMC is modified by training the non-coding Markov chain to explain every codon transition in both coding and non-coding regions in all frames. This creates a uniform non-coding model that, by design, does not possess a 3-base periodicity in either coding or non-coding regions. This allows the log-likelihood ratio $CMC(S) = \log \frac{\Pr(S|CMC_C)}{\Pr(S|CMC_N)}$ to preserve the 3-base periodicity in coding-regions.

The DiCodon Periodicity for a sequence is calculated by computing the CMC score for all three reading frames of the sequence and computing their inter-frame periodicity as in Equation 3.1. Because of the circular symmetry of the inter-frame periodicity computation, the resultant score is agnostic about the correct reading frame; it is purely a function of inter-frame scores for the CMC metric. Therefore, it can provide an orthogonal line of evidence for classifying a sequence as coding or non-coding.

$$CMC3(S) = x_1^2 + x_2^2 + x_3^2 - x_1x_2 - x_2x_3 - x_3x_1, \quad x_i = CSM(S_{FRAME_i})$$

3.2.2 Codon Composition Metric (CCM)

The codon composition metric (CCM) attempts to model the difference in codon composition in coding and non-coding sequences as directly as possible. Given a DNA sequence with a reading frame indicator, the number of occurrences of each of the 64 codons is counted (in-frame) and normalized by the number of codons in the sequence to create the codon composition distribution (CCD) feature vector. This 64-dimensional vector is then projected onto a discriminant vector to compute the codon composition metric score for the sequence.

The discriminant vector is a parameter of the metric and is trained by linear discriminant analysis (LDA) on a training set of sequences [30]. First the CCD for every sequence in a coding and non-coding training set of sequences is computed. The CCDs are treated as random variables from either the coding or non-coding class. The discriminant vector is then computed by finding a vector that maximizes the separation between classes when the CCDs are projected onto it.

The codon composition metric is similar to the Z curve metric, but without a dimensionality reduction or mono- or di-nucleotide frequencies. In Chapter 4, I demonstrate that although the codon composition metric uses less information than the Z-curve, it is more accurate when applied to alignments of sequences.

3.3 Novel Discriminative Phylogenetic Metrics

In this section, I present novel methods of discriminating between coding and non-coding sequences by exploiting the evolutionary bias in coding regions. The novel metrics presented, use discriminative models of observed sequence evolution informed by the phylogenetic relationship between the species.

3.3.1 Phylogenetic Codon Evolution

It can be shown that by computing the likelihood ratios of pair-wise codon substitutions under the coding and non-coding model, the Codon Substitution Metric (Section 3.1.2) computes a measure of the posterior likelihood of the sequence belonging to a protein coding region of the genome. As alignments of multiple genomes become available, we are able to observe alignments of codons in a large number of species. We would like to, therefore, be able to model the likelihood of observing certain codons in the informant species given the codon in the target species for both coding and non-coding regions, thereby discriminatively modeling the phylogenetic evolution of codons.

Consider the phylogenetic tree relating Humans and the three mammals Chimpanzee, Mouse and Dog as shown in Figure 3.3. Given an alignment of an exon in human, mouse, dog and chimpanzee, for each codon H in human, we wish to model,

$$\Pr(CMD|H, Coding)$$

where C , M and D are the observed aligned codons in Chimp, Mouse and Dog to codon H in Human.

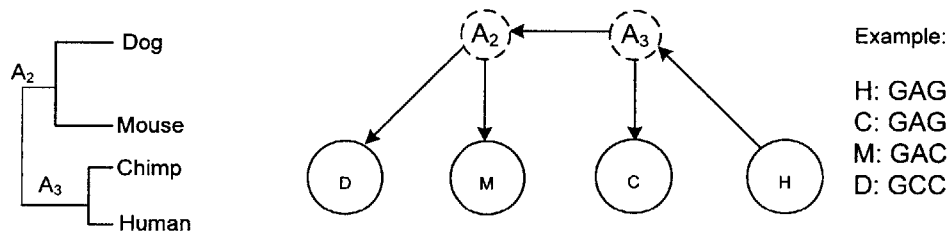


Figure 3.3 A codon substitution in the phylogenetic tree relating Human, Mouse, Dog and Rat rooted at Human

If there is nothing known about the evolutionary relationship between the species, a reasonable assumption would be to treat the codon substitutions independently and model each substitution with the observed Codon Substitution Matrix computed for each species.

$$\Pr(CMD|H, Coding) = \Pr(C|H, Coding) \Pr(M|H, Coding) \Pr(D|H, Coding)$$

$$\log \Pr(CMD|H, Coding) = CSM_{h,c}(H, C) + CSM_{h,m}(H, M) + CSM_{h,d}(H, D)$$

where $CSM_{h,i}$ is the codon substitution matrix (of log probabilities) between human and informant i . Such a model will be referred to as a “**Summed-Pairwise Codon Substitution Model**” and the resultant discriminative metric a “**Summed-Pairwise Codon Substitution Metric (PW-CSM)**”.

However, the substitution of codons between species is not independent. We do, in fact, know the phylogenetic tree relating the species in the alignment. The model should therefore incorporate that information to compute the joint probability of observing an alignment of informant codons to the target codon.

Given a phylogenetic tree, the probability of a codon at a node in the tree depends only on the codon in its ancestor node. Therefore, we can model the evolution of codons as a Bayesian network with codon substitution matrices forming the Conditional Probability Tables (CPT) on every branch. Since, only the annotation of the target codon is assumed, I choose the topology of the Bayesian network to be that of the phylogenetic tree relating the species rooted at the target genome. The nodes at the leaves are therefore the informant species and the internal nodes represent ancestors whose sequences are not observed.

Under the standard Markov model of molecular evolution [13], there is a non-zero probability of mutation in every codon during DNA replication. It is this process that results in an observable codon substitution matrix after thousands of generations. Therefore each codon substitution matrix is the result of a *unit codon substitution matrix* raised to a power proportional to the distance between the two species. Therefore, by observing codon substitution matrices between the target genome and each of the informants, one can estimate the unit CSM,

$$\mathbf{u}_C = \frac{1}{N} \sum_{k=1}^N (CSM_{T,I_k})^{\frac{\alpha}{d(T,I_k)}}$$

Equation 3.2

where CSM_{T,I_k} is the observed CSM between the target species and the k^{th} informant, $d(T,I_k)$ is the branch length between the target and informant and α is a parameter representing the length of the branch of \mathbf{u} . From the unit CSM, the CPTs of each branch can be computed by raising \mathbf{u} to a power proportional to the length of that branch.

$$C_{(n_1, n_2)} = (\mathbf{u}_C)^{\frac{d(n_1, n_2)}{\beta}}$$

Equation 3.3

where β is a conservation parameter.

Therefore, given observed CSM's between a target species and each informant species, as well as the phylogenetic tree relating the species, a Bayesian codon evolution network (CEN) can be fully specified by Equations 3.1 and 3.2, for some choice of parameters α and β . Figure 3.4 depicts the formation of the Bayesian network from a given tree topology and observed CSMs for coding regions.

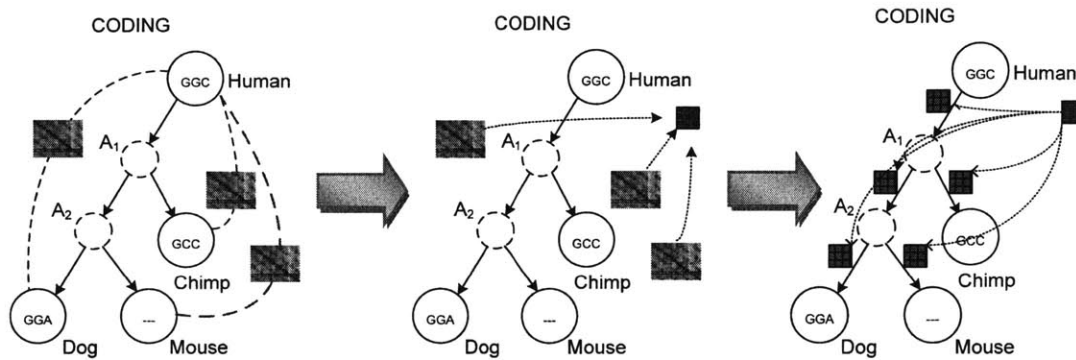


Figure 3.4: The estimation of CPTs of the Bayesian Evolution Network given the observed CSMs for each informant with the target genome.

Computing $\Pr(\{I\}|T, BNET)$

Given a target codon T , and a set of informant codons $\{I\}$, the conditional probability $\Pr(\{I\}|T, Coding)$ given the Bayesian network can be computed by summing the total joint probability of all the nodes (hidden and observed) over all the possible values of the hidden nodes. Because of the special structure in the network (each node having only one parent and the CPT along each branch having the same dimension), the conditional probability can be efficiently estimated with a recursive algorithm similar to Felsenstein's pruning algorithm [12]. Informant nodes (at the leaves) that are either unaligned or contain gaps are also treated as missing data and summed out. My algorithm for efficiently computing the conditional likelihood of informant codons given a target codon and Bayesian network is provided in Appendix A.

The Phylogenetic Codon Evolution Metric

The discriminative phylogenetic codon evolution metric trains two Bayesian codon evolution networks from observed coding and non-coding pair-wise codon substitution matrices. The conditional probabilities $\Pr(\{I\}|T, Coding)$ and $\Pr(\{I\}|H, Non - Coding)$ are computed independently for every column of codons in the alignment, and the phylogenetic codon evolution score is computed as the sum of the log-likelihood ratios at each alignment position.

$$PhyloCEN(\{I\} | T) = \sum_{j=1}^n \frac{\Pr(\{I_j\} | T_j, Coding)}{\Pr(\{I_j\} | T_j, NonCoding)}$$

where $\{I_j\}$ is the column of informant codons at triplet position j in the sequence.

Estimating the Parameters

At first it may seem unnecessary to allow two parameters for normalizing the branch lengths in the estimation of the unit CSM and the branch CPTs. Indeed, the intuition that α should equal β is true under a generative model. However, with the goal of the metric being discrimination between coding and non-coding, a different choice for the parameters may be more suitable.

Under a maximum likelihood objective function, α and β are chosen to maximize the probability of generating the informant codons given the target codon and the Bayesian network in protein-coding regions (and similarly for the non-coding network).

$$(\hat{\alpha}, \hat{\beta}) = \arg \max_{(\alpha, \beta)} \Pr(\{I\} | T, Coding)$$

While the above objective function ensures the model fits the data very well, it usually does not perform very well on the discrimination between coding and non-coding sequences. Therefore, to maximize discrimination the following alternative objective function is used,

$$(\hat{\alpha}, \hat{\beta}) = \arg \min_{(\alpha, \beta)} \Pr(\mathbf{s} \text{ is misclassified}) \quad \forall \mathbf{s} \in \text{Training set}$$

which is commonly referred to as the Minimum Classification Error criterion. Experimentally, it was found that the values of α and β that minimize the classification error differ significantly from the maximum likelihood estimates, and also from each other.

3.4 Integrated Metrics Incorporating Sequence and Evolutionary Signatures

The Bayesian Codon Evolution Network model proposed in Section 3.3.1 is a purely evolutionary metric for discriminating between coding and non-coding sequences. When scoring a sequence, it treats each column of codons in the alignment independently, and simply multiplies the probabilities of the observed informant codons given the target codon,

$$\Pr(\mathbf{I}_1 \mathbf{I}_2 \mathbf{I}_3 \mathbf{I}_4 \dots | T_1 T_2 T_3 T_4 \dots, \text{Coding}) = \prod_{i=1}^n \Pr(\mathbf{I}_i | T_i, \text{Coding})$$

where \mathbf{I}_i is a vector of aligned codons at position i .

The model is therefore agnostic about the sequence of codons in the target genome. However, it is widely accepted that there exist biases in the sequence and composition of codons in protein-coding sequences that alone serve as discriminative metrics for classifying coding and non-coding sequences. Therefore, I augment the codon evolution network to include a model of the target codon sequence thereby creating a model of the joint probability of all codons in the sequence alignment.

$$\begin{aligned} \Pr(\mathbf{I}_1 \mathbf{I}_2 \mathbf{I}_3 \mathbf{I}_4 \dots | T_1 T_2 T_3 T_4 \dots, \text{Coding}) &= \prod_{i=1}^n (\Pr(\mathbf{I}_i | T_i, \text{Coding})) \Pr(T_1 T_2 T_3 T_4 \dots, | \text{Coding}) \\ &= \Pr(\mathbf{I}_1 \mathbf{I}_2 \mathbf{I}_3 \mathbf{I}_4 \dots, T_1 T_2 T_3 T_4 \dots | \text{Coding}) \end{aligned}$$

I propose a new metric that uses a single-sequence Codon Markov Chain (CMC) to model the codon sequence in the target genome, resulting in a **Dynamic Bayesian Model of Codon Sequence and**

Evolution (DBN-CSE). The model is depicted in Figure 3.5 for an alignment of the target sequence with three informants. Using the CMC for modeling the target codon sequence has the advantage that training of the Dynamic Bayesian Network is unnecessary. A trained CMC model for the target sequence can be easily incorporated into the Bayesian architecture by simply adding the log-likelihoods. The DBN-CSE score is then given by,

$$\begin{aligned}
 DBNCSE(\{\mathbf{I}\}|\{T\}) &= \log \frac{\Pr(\{\mathbf{I}\},\{T\} | Cod)}{\Pr(\{\mathbf{I}\},\{T\} | Non-Cod)} = \log \frac{\Pr(\{\mathbf{I}\}|\{T\}, Cod)}{\Pr(\{\mathbf{I}\}|\{T\}, Non-Cod)} + \log \frac{\Pr(\{T\} | Cod)}{\Pr(\{T\} | Non-Cod)} \\
 &= PhyloCEN(\{\mathbf{I}\}|\{T\}) + CMC(\{T\})
 \end{aligned}$$

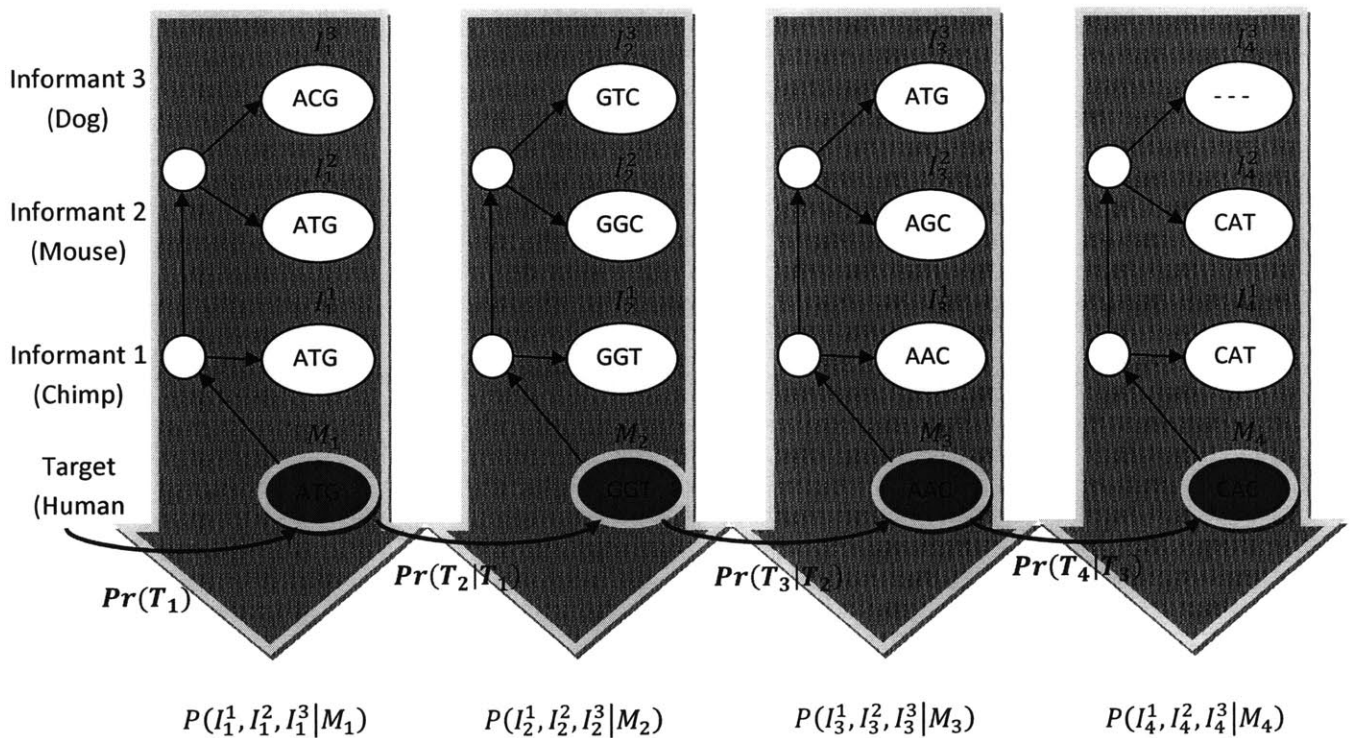


Figure 3.5: The Dynamic Bayesian Network of Codon Sequence and Evolution Model evaluating an alignment of four codons in four species.

4 Comparative Evaluation of Metrics on *de novo* Exon Classification

In Chapter 2, I described the ideal properties of a feature function in a linear chain semi-Markov conditional random field for automatic gene annotation. One of the most important characteristics is its ability to provide strong discrimination between protein coding and non-coding segments. Furthermore, because in unrestricted gene annotation, there can potentially be a large number of possible segmentations, only a few of which are real exons, a desirable metric feature function must provide very high exon classification accuracy.

In this chapter, we evaluate and compare the performance of the existing and novel metrics described in Chapter 3 on the classification of exons (protein-coding sequences) from the genome of the Fruit Fly *Drosophila Melanogaster* using whole genome alignments of 12 *Drosophila* [31], [7]. Analyzing classification performance also decouples the performance of the metrics from the performance of the CRF segmentation algorithm and any other splice-detection features that may be necessary to determine accurate exon boundaries in unrestricted genome annotation.

In Section 4.1, I describe the dataset of protein-coding and non-coding sequences used for the evaluation of the metrics and the methods used to quantify classification performance. In Sections 4.2, 4.3 and 4.4, I will analyze the performance of single-sequence metrics, single-sequence metrics extended to alignments and phylogenetic metrics respectively.

4.1 Protein-Coding Sequence Classification Data Set

The target genome for gene predictions was chosen to be the well studied fruit fly *Drosophila melanogaster*. With the recent sequencing of 11 related species, twelve-species whole genome alignments of *D. melanogaster* and related *Drosophila* is currently the largest alignment of animal genomes available for analysis [7]. Also, being one of the most well studied organisms, the annotations of the genes in *D. melanogaster* are comparatively more reliable than those of other organisms [32], thereby providing a more accurate data set for training and evaluating the metrics. Furthermore, studies have shown that almost 77% of disease genes in Human have strong matches in the genome of the Fly [33], further justifying its suitability as a model organism.

The phylogenetic tree relating the twelve *drosophila* species is shown in Figure 4.1. Using the FlyBase gene annotations [34] of the *melanogaster* genome, 10,722 exons were chosen randomly and segmented to create the protein coding sequences in the data set. 39,181 non-coding sequences were chosen randomly from the annotated non-coding regions of the genome. The non-coding segments were chosen to have the same overall length distribution as the exons to prevent length bias. When contiguous non-coding sequences of a certain length were not available in the genome, non-coding segments from different regions in the genomes were concatenated to create the sequences of the desired length. Furthermore, in frame stop codons were removed from the non-coding sequences to ensure an open reading frame in all coding and non-coding sequences.

The statistics of the data set are summarized in Table 4.1. The histogram of length distribution and conservation are shown in Figure 4.2 and Figure 4.3. The number of bases aligned per nucleotides of the target sequence is significantly higher in the protein coding sequences than in the non-coding sequences, with 7838 protein coding sequences having over 10.5 bases aligned on average in contrast to

1603 non-coding sequences. This gives us a simple baseline classification sensitivity of 73% with a false positive rate of 3.15%, and an overall average classification error of 15%.

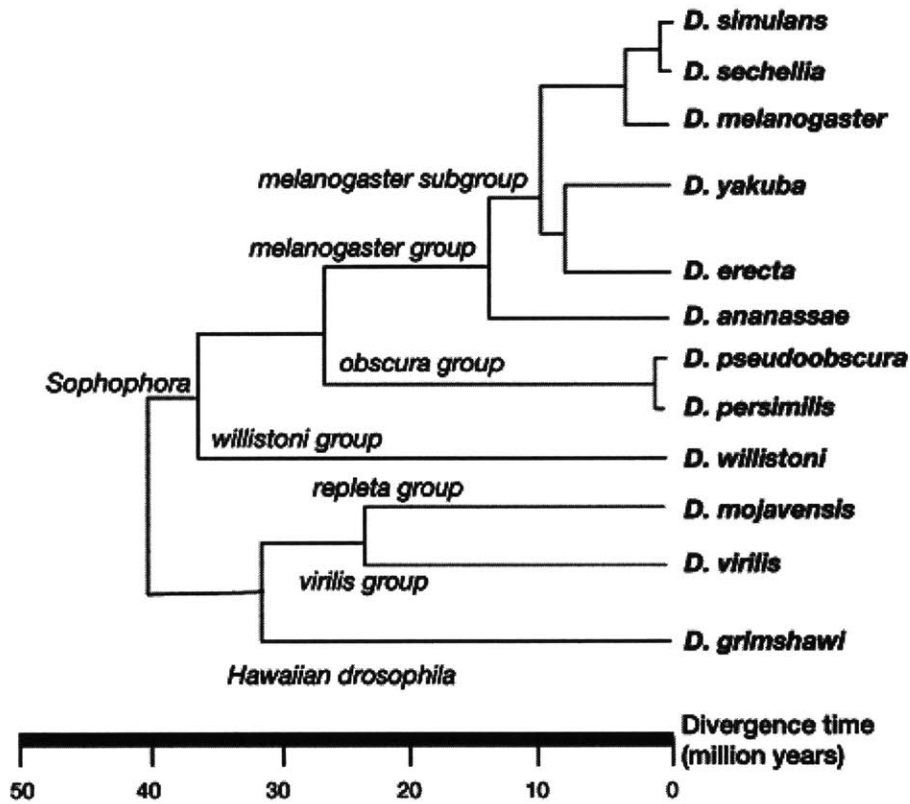


Figure 4.1 The phylogenetic tree relating *D. melanogaster* to its 11 sequenced relatives (informants)

Table 4.1 Statistics of the Fly exon data set used to evaluate the metrics

	Exons (Coding Sequences)	Non-Coding Sequences
Number of Sequences	10722	39181
Average Length	405	404
Average GC Content	0.522	0.416
Average Number of Bases Aligned (per target Nucleotide)	10.44	7.81

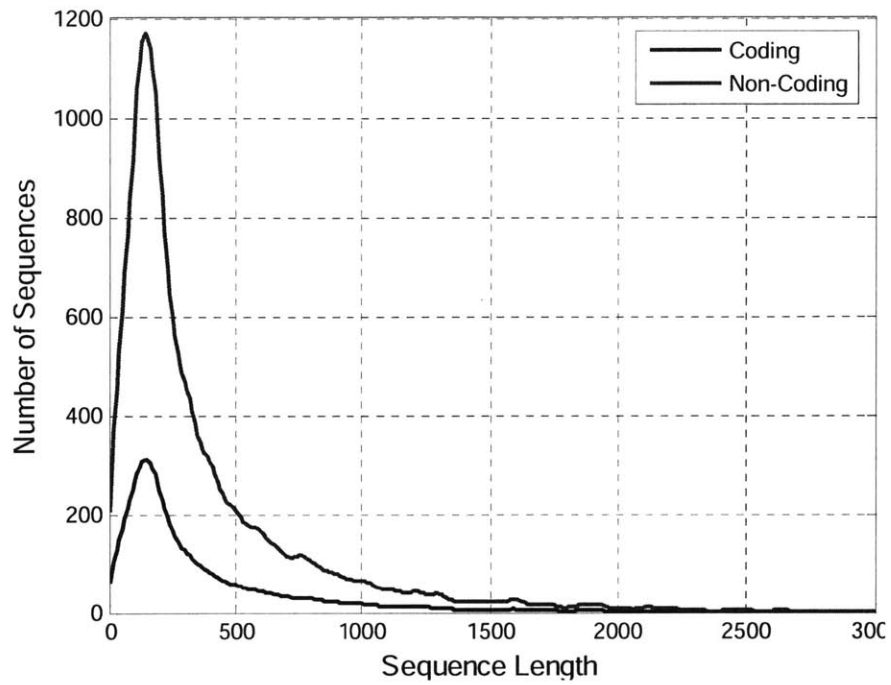


Figure 4.2: Length distribution of coding (blue) and non-coding (red) sequences in the data set.

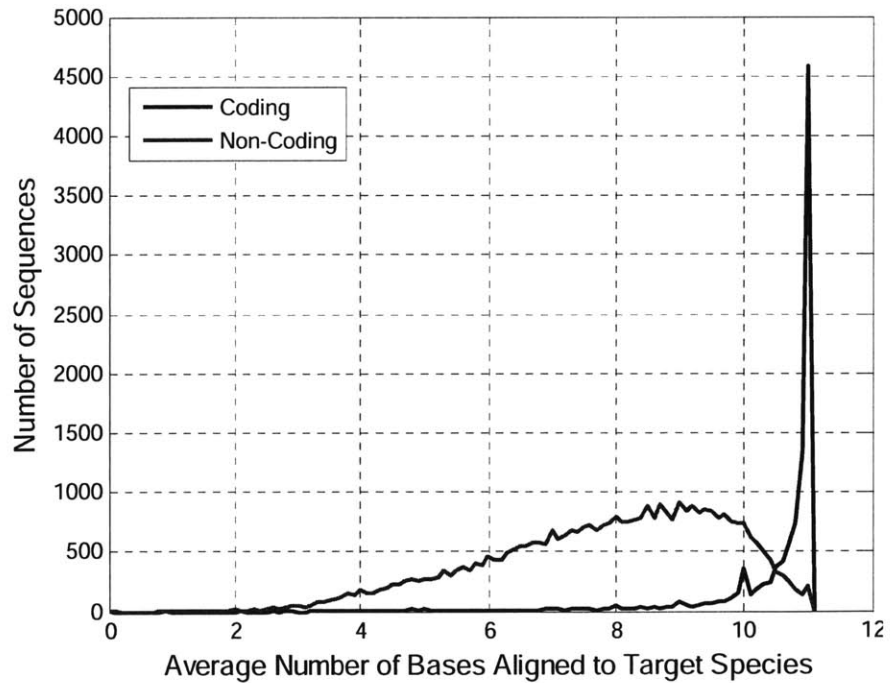


Figure 4.3: Average number of informant nucleotides aligned to the nucleotides in the target sequence (*D. Melanogaster*) in coding (blue) and non-coding (red) sequences.

Quantifying Metric Classification Performance:

Given any set of positive (P) and negative (N) examples, a classifier's performance is measured by the following quantities:

Number of True Positives (PP)	The number of positive examples correctly classified as positive
Number of False Positives (NP)	The number of negative examples incorrectly classified as positive
Number of False Negatives (PN)	The number of positive examples incorrectly classified as negative
Number of True Negatives (NN)	The number of negative examples correctly classified as negative
True Positive Rate, Sensitivity (Sn)	The fraction of positive examples correctly classified as positive $S_n = \frac{PP}{\text{Number of Actual Positive}} = \frac{PP}{PP + PN}$
True Negative Rate, Specificity (Sp)	The fraction of negative examples correctly classified as negative $S_p = \frac{NN}{\text{Number of Actual Negative}} = \frac{NN}{NN + NP}$
False Positive Rate, (Fpr = 1-Sp)	The fraction of negative examples incorrectly classified as positive $F_{pr} = \frac{NP}{\text{Number of Actual Negative}} = \frac{NP}{NN + NP}$
Average Error Rate, (AER)	$AER = \frac{1 - S_n + F_{pr}}{2}$
Precision, (Pr)	$P_r = \frac{PP}{\text{Number of Actual Positive}} = \frac{PP}{PP + NP}$

All of the discriminative metrics presented, map a sequence of DNA or an alignment of sequences of DNA to a real valued score that is strongly correlated with the conditional probability of the sequence being a protein coding sequence. Since the scores need not be actual probabilities, we will term them *protein-coding potential*. To classify a sequence, its protein-coding potential must be compared to a threshold (a parameter of the metric) and determined to be coding if the potential is greater than the threshold. Therefore, the classification performance of a metric is evaluated with a Receiver Operating Characteristic (ROC) curve, a continuous curve that plots values of the true positive rate against the false positive rate for different values of the cutoff. The curve represents all the points of operating classification performance that are achievable by the classifier on the data set. The ROC curve for a

perfect classifier (which produces no false positives and only true positives) is a single point at (0, 1). An example of an ROC curve is shown in Figure 4.4 with important quantities highlighted.

The following two quantities are reported to measure the performance of a metric as represented by the ROC curve:

Minimum Average Error Rate (MAE)	The average error rate that is achieved at the point on the ROC curve with the shortest “city-block” distance to the point of perfect classification (0,1), usually represented by a red square on the curve.
Area Above the Curve (AAC)	The area above the curve is another measure of error evaluated using all the points on the ROC curve. A perfect classifier has an AAC of 0 while a random classifier has an AAC of 0.5

Both MAE and AAC are a measure of the classification error, and a metric with lower values of MAE and AAC is more accurate. Both quantities are reported because either one may not always be strictly better than the other as a measure of classification performance. For example, because of restrictions on either the lower bound of S_n or the upper bound of F_{pr} , the MAE point of operation may not be achievable. In such a case AAC might be a better measure to compare classifiers. For the classification goal, Precision is not reported as it is dependent on the relative number of negative examples, whereas F_{pr} is not. The point of minimum classification error is also not reported because in all cases, due to the large number of non-coding sequences, it tends to favor points with very low F_{pr} at the cost of low S_n .

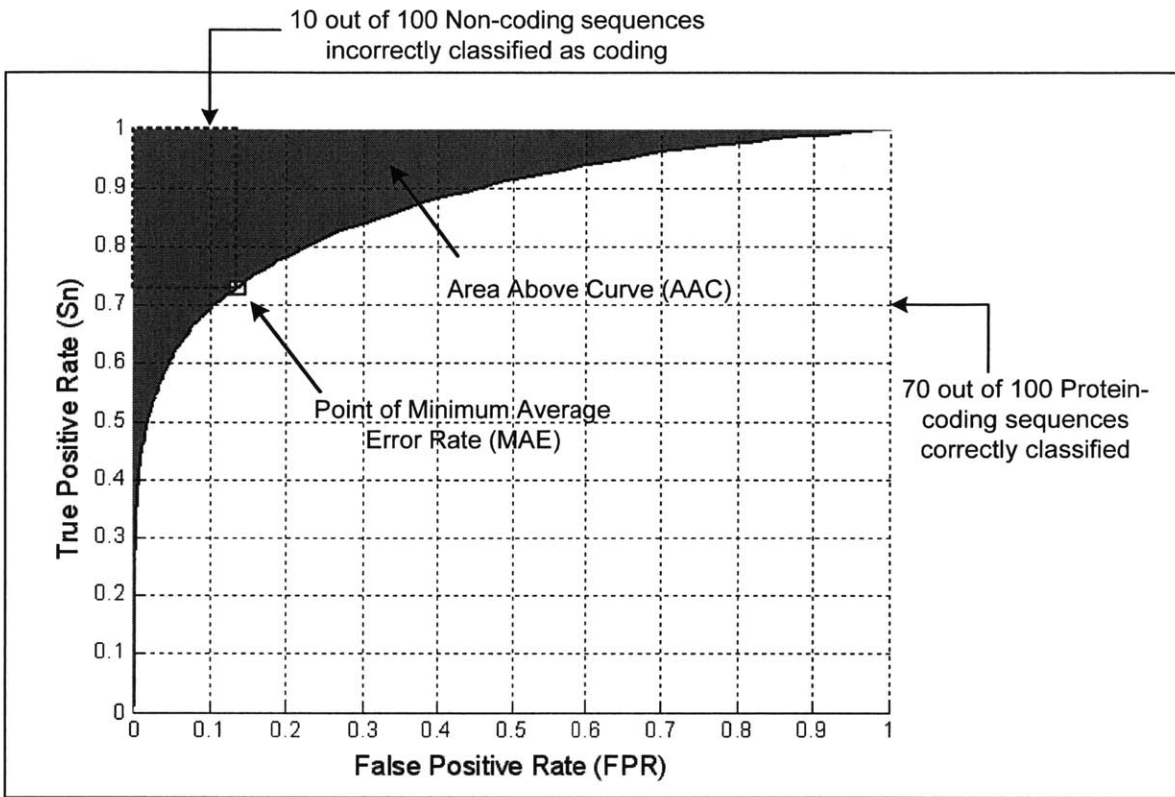


Figure 4.4: An example ROC curve for the classification performance of a metric.

4.2 Comparative Analysis of Single-Sequence Metrics

In all of the following evaluations, the metrics were used to score each of the 49903 sequences in the database. The ROC curves, minimum average error (MAE) and area above curve (AAC) quantities for the protein-coding potential score for each metric was reported. Four-fold cross validation was used to evaluate any metric that required training (ie. each quarter of the data set was classified by a different model trained on the remaining three quarters of the data).

Figure 4.5 shows the ROC curves for some of the existing (baseline) single-sequence metrics. The best performing metric is the Z-Curve Score at low MAE error rates. The Codon Markov Chain (CMC) performs better at higher values of sensitivity or lower false positive rates. The Codon Markov Chain

also outperforms the Interpolated Markov Model (IMM) at all points along their respective curves. Given that both metrics are Markov models of sequence structure, this disparity suggests that a simple first order model of codon sequence structure is better suited for discrimination between coding and non-coding sequences than a multiple order nucleotide model.

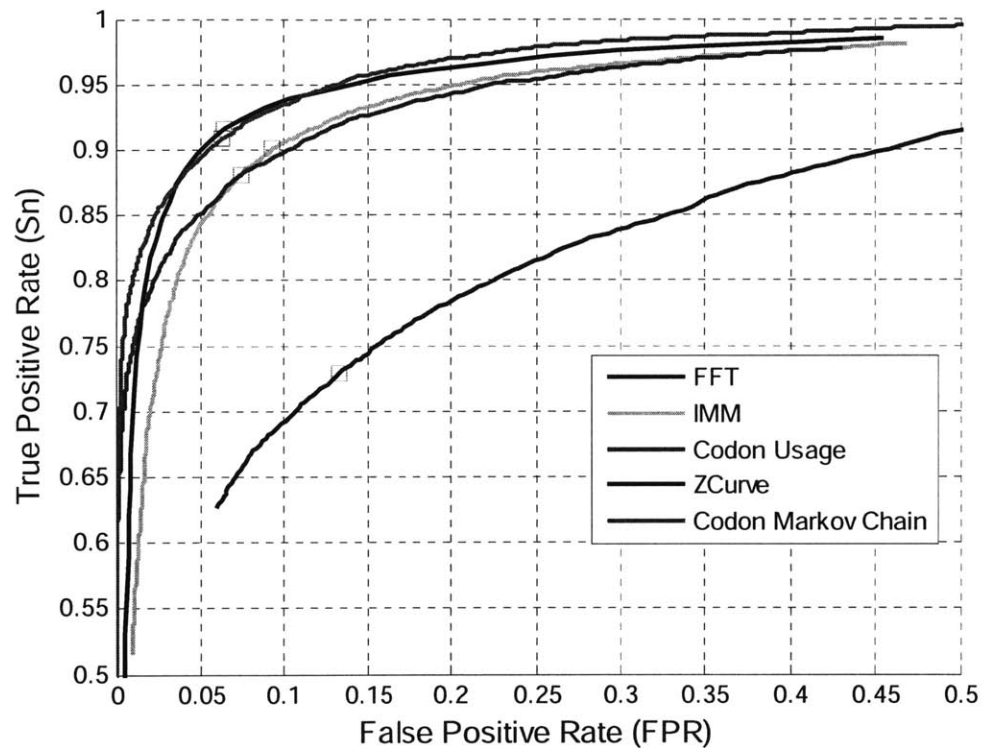


Figure 4.5: ROC curves for Baseline Single Sequence metrics.

The performance of the DiCodon Periodicity metric is shown in Figure 4.6. The DiCodon Periodicity, represented as CMC3, is also always superior to the FFT metric, suggesting that the inter-frame periodicity property is an important signal, especially when applied to numerical transformations of the genetic sequence that are highly frame-specific for coding regions but not so for non-coding regions. It should be noted that the CMC non-coding model is trained differently for discrimination based on in-frame log-likelihood-ratio (CMC), than for discrimination based on inter-frame periodicities (CMC3), as described in Section 3.2.1. Also, notice that the combination of CMC and CMC3, which is labeled CMC+3

has a lower MAE error rate than CMC alone, implying that the inter-frame periodicity of the CMC log-likelihoods incorporates different sequence biases than the signal itself, and that their combination can lead to superior discrimination.

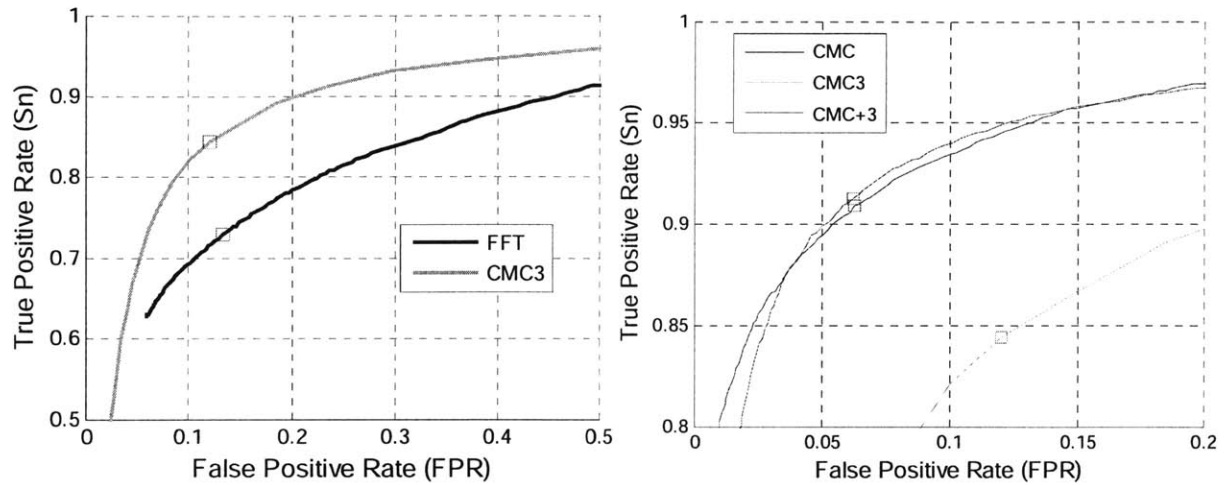


Figure 4.6 Comparison of the ROC curves of the DiCodon Periodicity metric

Lastly, the performance of the Codon Composition Metric (CCM), Codon Usage Metric and Z-Curve is shown in Figure 4.7. CCM performs at much higher sensitivity and specificity than the Codon Usage Metric, but is not as accurate as the Z-Curve method. This implies that mono-nucleotide and di-nucleotide frequencies provide added dimensions of stronger discrimination for single sequence classification. The performance of all the single sequence metrics are summarized in Figure 4.7 and Appendix B.

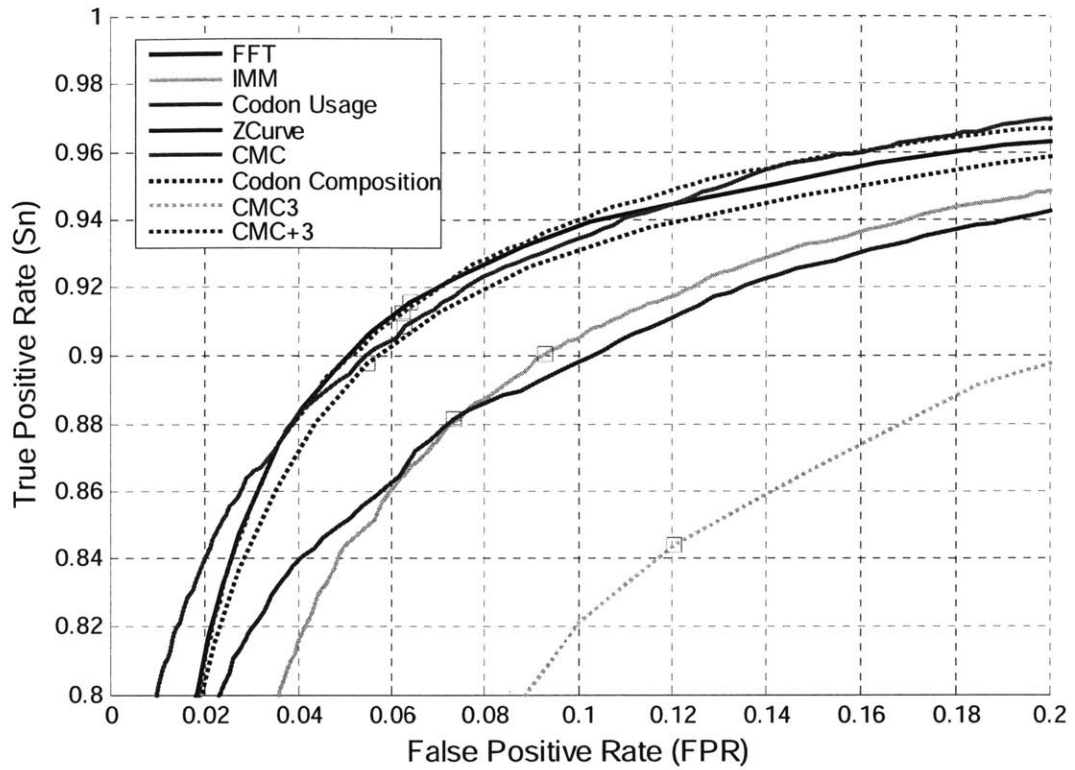


Figure 4.7 ROC curves comparing the performance of baseline single species metrics. The novel metrics are shown in dotted lines.

4.3 Comparative Analysis of Sequence-Based Alignment Metrics

In this section, the sequence-based metrics are extended to alignments of sequences. In most cases, the extension is deeper than a simple sum of scores of each informant species in the alignment. The extension methodology of each metric and their ROC curves are presented below.

The FFT metric for alignments was computed by averaging the indicator sequences of each nucleotide over the number of species in the alignment. The periodicity was then calculated on the indicator sequences in the usual way. While the FFT metric showed an improvement in classification by using the alignment, the improvement was not as pronounced as that of the DiCodon Periodicity.

The DiCodon Periodicity for the alignment was computed by using the Codon Markov Chain models to compute the log likelihood ratios for all frames in the target and informant species while ignoring gaps.

The resulting log-likelihoods were averaged over the number of species to compute the frame specific log-likelihoods. The inter-frame periodicity was then computed normally. The CMC score for the alignment was computed slightly differently. Instead of removing gaps from the informant species, the DiCodon log-likelihoods were computed for all complete in-frame di-codon subsequences. The score of the alignment was then computed by summing all the in-frame log-likelihood ratios. Figure 4.8 shows that the DiCodon Periodicity of Alignments (CMCA3) metric is a more accurate classifier than the CMC Alignment (CMCA) score. Furthermore, the combination of the two metrics shows a significant boost in performance at all points along the ROC, suggesting that the inter-frame periodicity and the codon sequence log-likelihood scores exploit different biases in the alignments of sequences leading to superior overall performance. The resulting metric is one of the most accurate sequence-based metrics producing a minimum average error rate of 4.5% (on average 4.5 out of every 100 sequences misclassified). It should also be noted that the training procedure is exactly the same as that of single-sequence classification. It was found that training different CMC models for each informant greatly increased the parameter space without substantially improving discrimination performance. Those results are not presented here.

Another attractive feature of the DiCodon metric is that the performance scales with the increase in the number of informants as shown in Figure 4.9. The dotted lines in the figure represent pair-wise alignments. The best pair-wise informant is the species *D. ananassae*, which is at an ideal evolutionary distance from the target species *D. melanogaster*. The accuracy of the DiCodon periodicity metric steadily increases as the number of informants is increased.

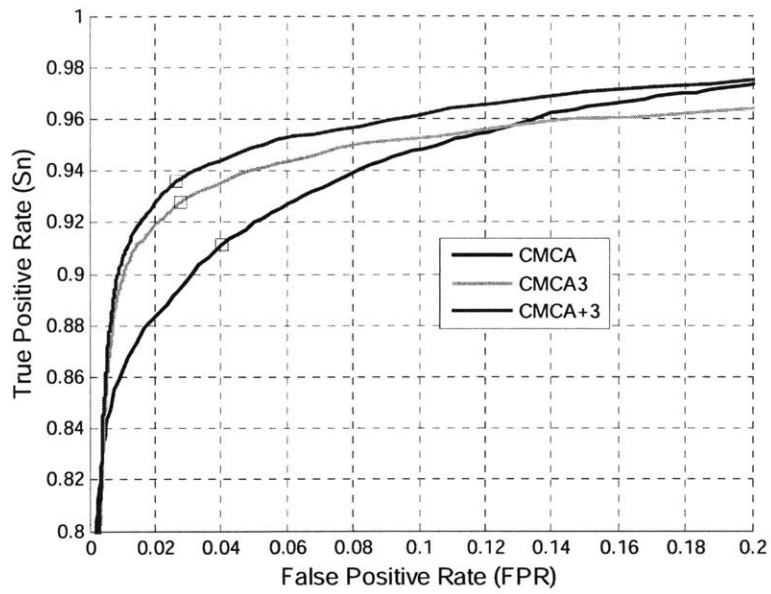


Figure 4.8: The performance of the DiCodon metrics on alignments of sequences. The combination of codon sequence bias as well as inter-frame periodicity results in superior classification.

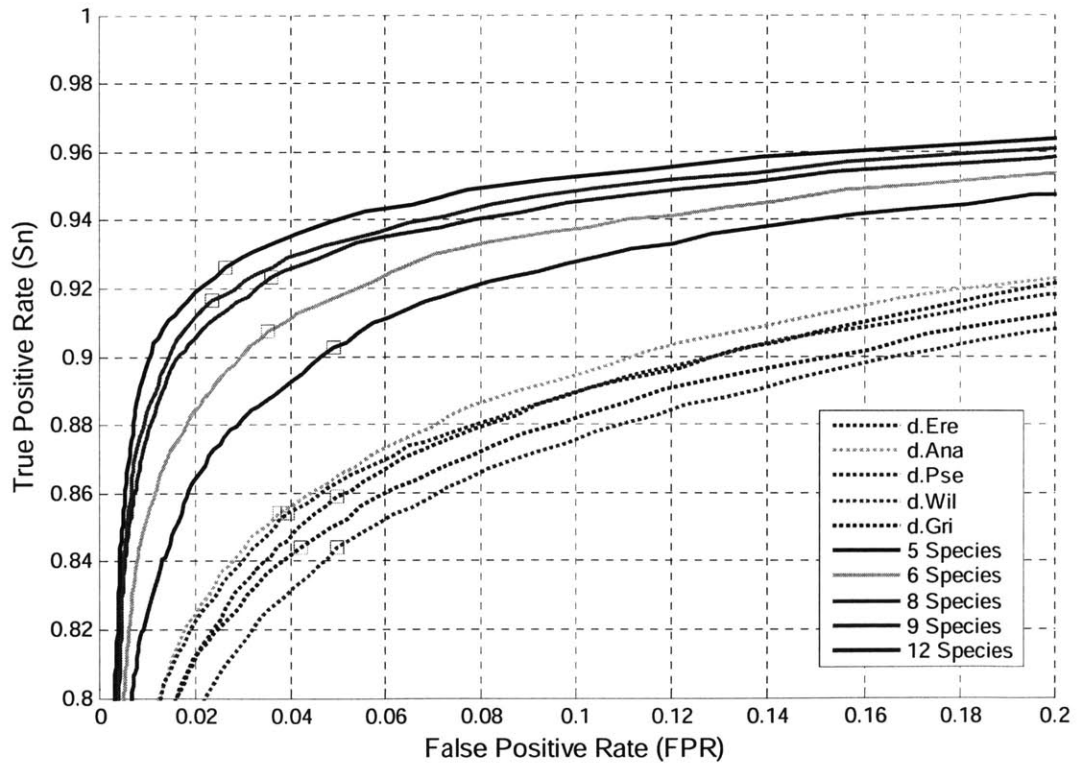


Figure 4.9: The increase in performance of the DiCodon Periodicity (CMC3) metric as the number of informant species is increased.

The ROC curves for the Z-Curve Alignment (ZCA) and Codon Composition of Alignment (CCMA) metrics are shown in Figure 4.10. In both cases, the sequences of the informants were treated as additional evidence used to gather the mono- and di-nucleotide statistics (ZCA) and the codon composition statistics (for ZCA and CCMA). The resultant multidimensional vectors were then projected onto their respective linear discriminant hyper-planes to compute the protein-coding potential scores. The performance of the codon composition metric is much higher than that of the Z-Curve metric for alignments of sequences, showing that the CCMA scales better with alignments of sequences than does the ZCA.

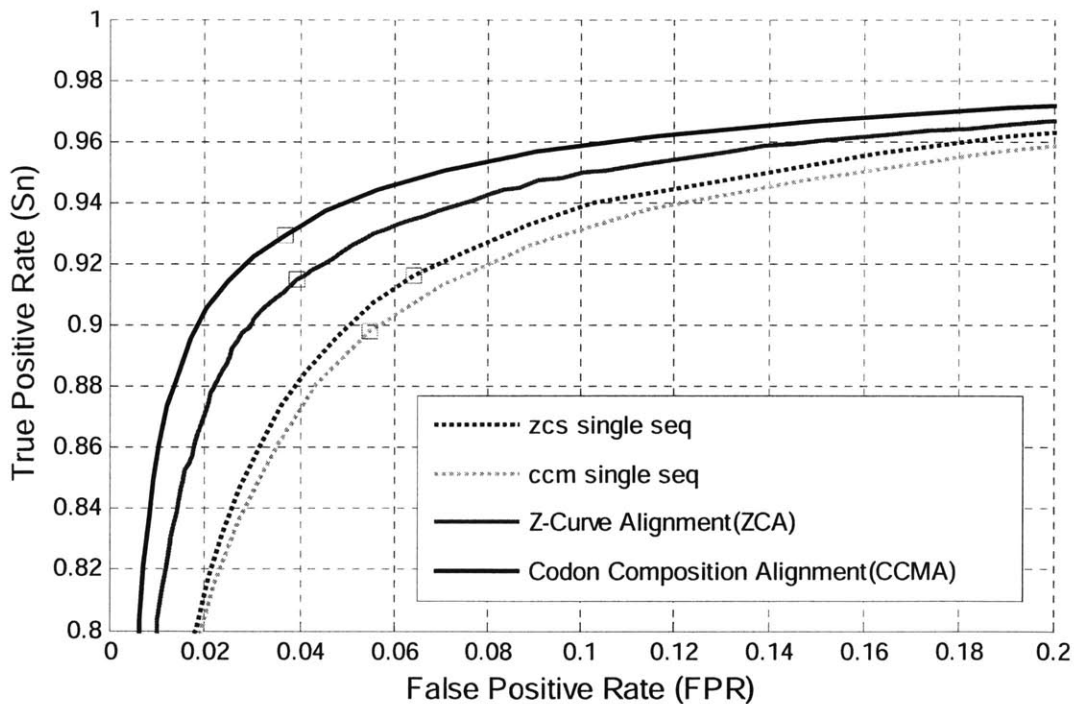


Figure 4.10 The ROC curves showing the greater accuracy of Codon Composition Metric for Alignments over the Z-Curve metric

Finally, I evaluate the conservation metric and its periodicity. The ROC curves for Alignment Conservation (CONS), Conservation Periodicity (CONS3) and the combination of the two (CONS+3) is shown in Figure 4.11. While the conservation of a sequence is a strong indicator of its protein-coding potential (the baseline CONS metric achieves 91.5% classification accuracy), the inter-frame periodicity

of conservation is often a better discriminator. Furthermore, the inter-frame information exploited by CONS3 when combined with CONS produces an even more accurate classifier (CONS+3) suggesting that the two signals capture slightly different protein-coding biases. The ROC curves for each sequence-alignment-based method is provided in Appendix B.

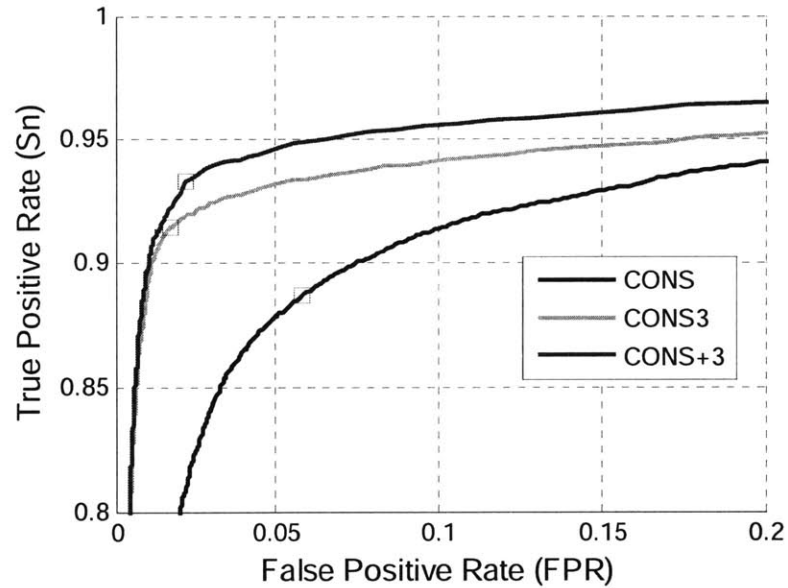


Figure 4.11: Comparative analysis of the baseline conservation metric, the inter-frame periodicity of conservation (CONS3) and their combination (CONS+3)

4.4 Comparative Analysis of Evolutionary Metrics

In this section we analyze the performance of the Phylogenetic Codon Evolution Network metric, and the Dynamic Bayesian Network of Codon Sequence and Evolution.

Training:

The Phylo-CEN was trained by first estimating the empirical Codon Substitution Matrices for each of the informants conditioned on the target genome, from both coding and non-coding sequences. All of the training was performed by four-fold cross validation. Therefore, four models were trained for each

quarter of the data set to ensure none of the sequences to be classified were seen in the training set of their respective model. Once the CSMs were obtained, the *rate-of-evolution* parameters α and β were estimated by training a number of different models for different values of α and β and comparing their discriminative performance on a small ‘development’ set (1000 coding and 4000 non-coding sequences). The choice of α and β that minimized the minimum average error (MAE) rate on the development set was chosen, thereby performing a modified Minimum Classification Error (MCE) parameter estimation. The table of values of Model Likelihood Error (Generative Parameter Estimation) and Minimum Average Classification Error (Discriminative Parameter estimation) are shown in Appendix B. Large values of $\alpha \approx \beta$, were found to produce the best model fit, or maximize the model likelihood of generating the data, but offer poor discrimination. The best choice of α and β , that minimized classification error was found to be,

$$\alpha = 0.024 \text{ and } \beta = 1$$

The choice of $\alpha = 0.024$ implies that the unit evolutionary CSM matrix determined from the pairwise CSMs has a time scale equivalent to that of the shortest branch on the phylogenetic tree relating the species. This seems counter-intuitive, as one would imagine the longest branch in the tree affording the best estimate of divergence, as is indicated by the model likelihood. However, since our goal is the discrimination between coding and non-coding sequences, we can allow α and β to take on values that make the codon evolution in coding sequences as explained by the coding Bayes net less likely if at the same time, it makes the codon evolution explained by the non-coding Bayes net much less likely. Since our metric score is the ratio of likelihoods and not the likelihoods themselves, we can get better performance by choosing α and β to maximize that ratio for coding sequences but not for non-coding sequence, which is what discriminative training achieves.

Once the parameter values were obtained, the phylogenetic codon Bayesian networks (PhyloCEN) were computed and employed to score each sequence in the test set as described in Section 3.3.1. The Bayesian networks were also combined with the single-sequence Codon Markov Chain models (CMC) to evaluate the Dynamic Bayesian Codon Sequence and Evolution Network (DBNCSE), as described in section 3.3.2.

The ROC curve for the PhyloCEN metric and the DBN-CSE metric is shown in Figure 4.12. They are compared to the baseline conservation metric (CONS) as well as the Pairwise-Summed Codon Substitution Metric (PW-CSM) and the Reading Frame Conservation Metric (RFC). PhyloCEN is seen to be clearly superior to the pairwise-CSM metric at all points on the ROC curve. Also, as expected, the DBN-CSE metric, being a metric that exploits both sequence and evolutionary biases, outperforms all other metrics at all points on the ROC. By incorporating phylogenetic information, the **PhyloCEN achieves a 15% relative reduction in error over PW-CSM**, and by incorporating both phylogeny and sequence structure, the **DBN-CSE achieves a 20% relative reduction in error over PW-CSM**.

Figure 4.13 demonstrates the improvement in accuracy of the PhyloCEN with increase in the number of informants. As expected, the performance scales appropriately with the number of informants and size of phylogenetic tree.

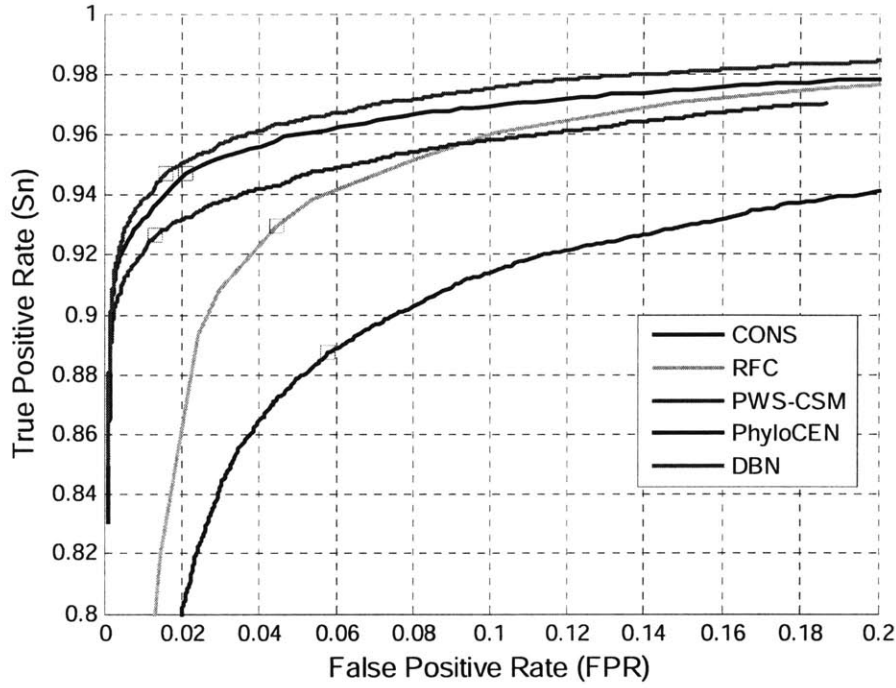


Figure 4.12: The ROC curves for the Pairwise-Summed Codon Substitution Metric (PW-CSM), the Phylogenetic Bayesian network of Codon Evolution (PhyloCEN) and the baseline Conservation metric (CONS).

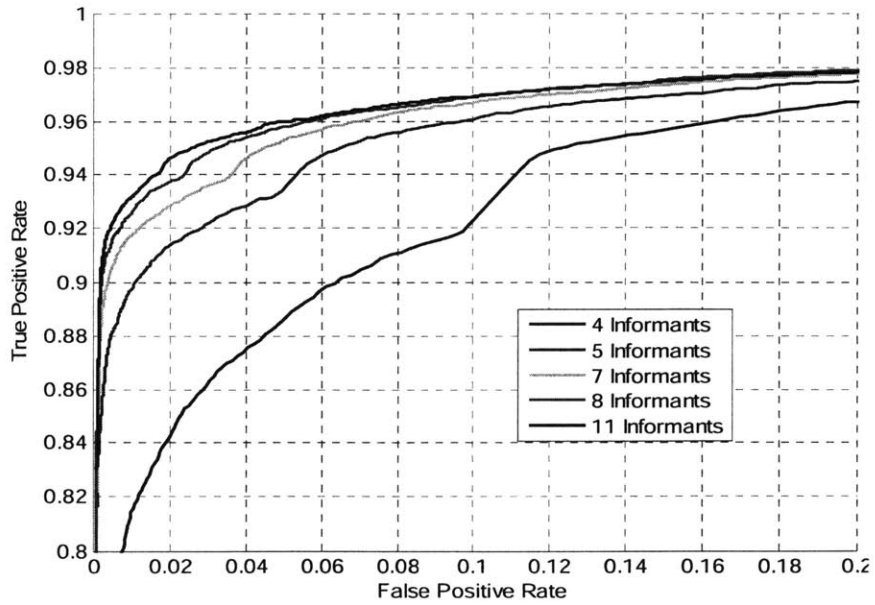


Figure 4.13: The increase in classification accuracy of the PhyloCEN metric as the number of informants is increased.

The performance of all the metrics is summarized in Figure 4.14. The novel metrics presented in this thesis are colored in darker colors. The novel metrics outperform most of the metrics in their class as well as over all classes. **The Dynamic Bayesian Network of Codon Sequence and Evolution is the single best performing metric with a Minimum Average Error of 3.45% and an AAC of 1.2%.**

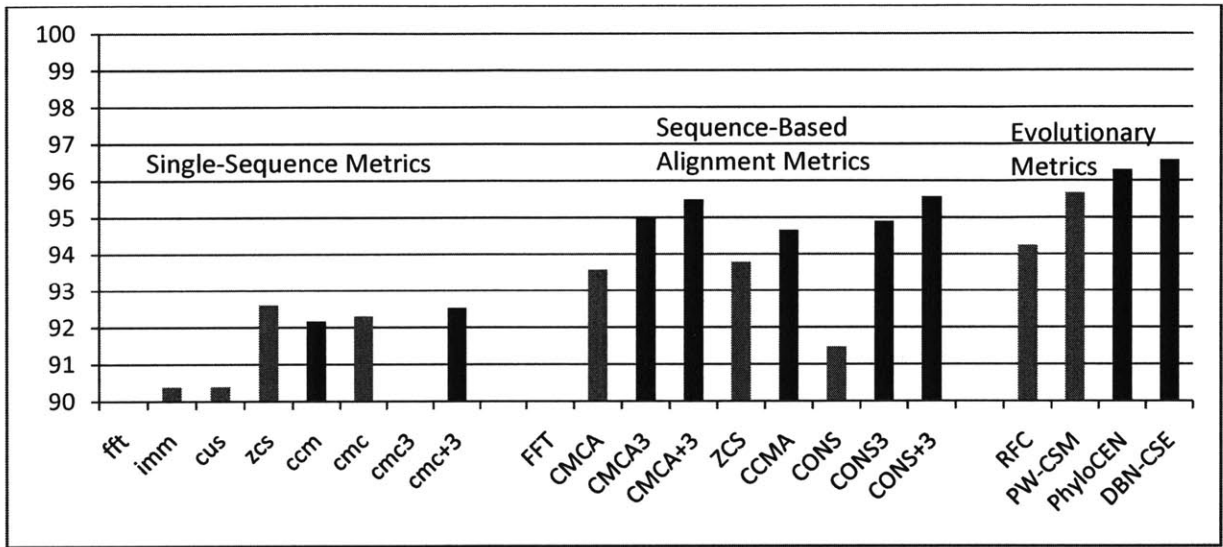


Figure 4.14: Comparative analysis of Novel Metrics (Dark Red) and Existing Metrics (Light Red). Accuracy is reported as 1-MAE.

5 Combinations of Metrics for Improved Classification

All of the metrics presented have been based on one or more of generally three different types of biases inherent in protein-coding regions of the genome. The first bias, coding sequence composition, is the bias in the relative occurrences of certain sequences or patterns of sequences in coding regions as a result of the genetic code. The second bias is an evolutionary pressure of constrained mutation in coding regions. The final bias is an inter-frame periodicity observed in many frame specific signals. It has already been shown that, in some cases, metrics that exploit different biases can be incorporated into a single metric that outperforms either component metric. For example, it was found that the protein-coding potential scores of DiCodon Periodicity (CMCA3) and CMCA could be combined to produce the CMCA+3 metric that outperformed either one. Another reason for exploring metric combinations is that in a Conditional Random Field architecture, many metrics can be used in conjunction with each other as feature functions, as shown in Figure 2.7. In such a framework, it may be advantageous to use metrics whose combinations produce classifiers that perform better than any of them individually. In this section, I investigate different methods of combinations of classifiers to achieve the best overall classification. To restrict the size of the combination space, I will investigate metric combinations of the following ten metrics: DBN-CSE, PhyloCEN, RFC, CONS, CONS3, CMCA+3, CCMA, CMC, IMM, ZCS, chosen from all three classes that exploit all three biases in protein-coding sequences.

The scores produced by the metrics are not generally required to be within a specific range of values. Therefore the scales and distributions of scores can vary greatly between metrics. The metric combination techniques discussed were found to perform very poorly if combining raw metric scores. Therefore, I implemented the following framework, which can be applied to combine any arbitrary set of metrics.

5.1 Posterior Probabilities Framework

While the metric scores represent the protein coding potential of a sequence, they are not true probabilities. Therefore, by observing the scores of a metric (M_1) for a training set of sequences belonging to the same class (coding or non-coding) the conditional probability density functions $P_{M_1}(x|Coding)$ and $P_{M_1}(x|NonCoding)$ can be estimated. Once the PDFs are known, the posterior probability can be computed by Bayes rule,

$$\Pr(Coding|x) = \frac{P_{M_1}(x|Coding) \Pr(Coding)}{P_{M_1}(x|Coding) \Pr(Coding) + P_{M_1}(x|NonCoding)(1 - \Pr(Coding))}$$

The posterior is usually computed by using log likelihoods of the PDFs,

$$\log \Pr(Coding|x) = -\log \left(1 + \frac{1 - \Pr(Coding)}{\Pr(Coding)} \exp \left(-\log \frac{P_{M_1}(x|Coding)}{P_{M_1}(x|NonCoding)} \right) \right)$$

With finite data, estimating the density functions can be quite difficult. Firstly, because of the ratio of probabilities in the previous equation, the density estimates need to be smooth. Secondly, for this method to be applicable to any metric, we cannot place a parametric assumption on the true distribution of scores. Thirdly, since the densities are estimated from a cross-validated training set, they must not assign zero probabilities to unseen scores. Therefore non-parameterized techniques such as histogramming cannot be used. The best technique for estimating densities that satisfied all these properties found to be kernel density estimation using Parzen windows [30]. In this technique, a small Gaussian shaped kernel is placed at all points corresponding to the metric scores in the training set. The distribution is then calculated by adding the contributions from all the kernels, resulting in a smooth probability density function.

The posterior distribution technique is used to compute the posterior probabilities for every metric for every sequence using four-fold cross validation. As a result each sequence is now represented by a length-10 feature vector of posterior probabilities which can then be used to classify it.

5.2 Linear Discriminant Analysis

Linear Discriminant Analysis is a classical technique of discriminating between a set of multidimensional real valued vectors by computing a linear discriminant function that maximizes the separation between the two classes [30]. Given a set of real valued feature vectors $\{X\}$ and their associated class labels $\{Y\}$, LDA computes the optimal linear discriminator Λ such that the projection of each feature onto the discriminator $\Lambda \cdot X = \sum_i \lambda_i X_i$ produces the maximum separation between the two classes.

One of the principal reasons for evaluating LDA combinations is that in the CRF architecture the overall score given to a segment is a weighted sum of the scores produced by the feature functions for that segment. Furthermore, by examining the discrimination vector, the contribution of each metric to the combination can be assessed.

The unit-length discriminant vector computed by LDA for the data set (averaged over the four cross-validated training sets) is shown in

Table 5.1. Notice that the weights assigned to the DBN-CSE, Conservation Periodicity and DiCodon Periodicity metrics is much higher than the rest of the weights. This is understandable because those are the most discriminative metrics that also use different types of biases. Also notice that the weight assigned to RFC is much lower than anticipated given its accuracy. The reason for this was found to be that, because the RFC score is a discrete number from 0 to 11, the kernel density estimate is very poor leading to very unreliable posterior probabilities. However, the metric was still used in the combination framework to test the framework's robustness.

The performance of LDA compared to the best performing metric (DBN-CSE) and other metric combinations is shown in Figure 5.1.

Table 5.1: The discriminant weights assigned to each metrics' posterior probabilities as discovered by LDA

Metric	Discriminant Vector Weight
DBN-CSE	0.569
PhyloCEN	0.275
CMCA+3 (DiCodon+Periodicity)	0.303
CONS3	-0.379
Cmc	-0.214
CCM (Codon Composition)	0.100
CONS+3	0.519
IMM	0.075
RFC	0.048
ZCS	0.182

5.3 Support Vector Machines

While LDA performs linear discrimination on the metric-posterior feature vectors, Support Vector Machines can map the features to a higher dimension to discover a linear separator that maximally separates the data. SVMs were trained using the SVM-Light package [35], using four-fold cross validation to avoid over fitting. The prediction confidence for each sequence was then used to compute the ROC curve for this combination method. The ROC curve is shown in Figure 5.1

5.4 Majority Voting

Being one of the simplest methods of combining classifiers without assuming any knowledge about the classifiers, majority voting was used to classify a sequence as coding if the number of metrics that classified it as coding was greater than a majority threshold. One advantage of majority voting is that unlike the other metric combination techniques, it uses the hard classification output of each metric (the class label) instead of the soft classification output (the metric score or posterior probability).

Therefore, the posterior probabilities do not have to be computed and cross validation becomes unnecessary. Such a technique may be useful if some of the metrics have PDFs that cannot be estimated reliably (such as RFC). The ROC curve for majority voting was created by varying the majority threshold, to compute different points of operation. The results from majority voting combination are shown in Figure 5.1.

5.5 Combinatorial Posterior Combinations

To evaluate the performance of every combination of metrics, the posterior probabilities were computed for all $2^{10} - 11$ possible combinations of metrics to find the combination that produced the lowest MAE and AAC on the training sequences.

$$\Pr(Cod|x_1, x_2, \dots, x_{10}) = -\log\left(1 + \frac{1 - \Pr(Cod)}{\Pr(Cod)} \exp\left(-\sum_i \log \frac{P_{M_i}(x_i|Cod)}{P_{M_i}(x_i|NonCod)}\right)\right)$$

The best performing combination was then used to classify the test sequences. This was done with four-fold cross validation across the data set. The combination of metrics that produced the lowest MAE and AAC was found to be,

Lowest MAE	Lowest AAC
DBN-CSE	DBN-CSE
PhyloCEN	PhyloCEN
CMCA+3	CMCA+3
CONS+3	CONS+3
IMM	ZCS
RFC	
ZCS	

The lowest MAE and AAC combination curves are shown in Figure 5.1. The error rates for all combinations of metrics are shown in Appendix C.

Figure 5.1 shows the ROCs from various methods of combinations. The ROCs for all methods turned out to be surprisingly similar. The accuracy of SVM was worse than LDA suggesting that the SVM models are more prone to overfitting (SVMs outperformed LDA on non-cross-validated data). The majority voting method performed surprisingly well, especially at high values of S_n (low majority threshold). Finally, the ROC curves for the posterior combination methods were found to be quite similar to those of LDA which is understandable given that the metrics given a high weight by the discriminant vector in LDA are the same as those chosen by the posterior-combination. **The best performing combination method was LDA with a relative reduction in classification error rate of 17% over the best performing single metric.** Since feature functions scores are linearly combined in the CRF, this result also further shows the suitability of metrics as feature functions.

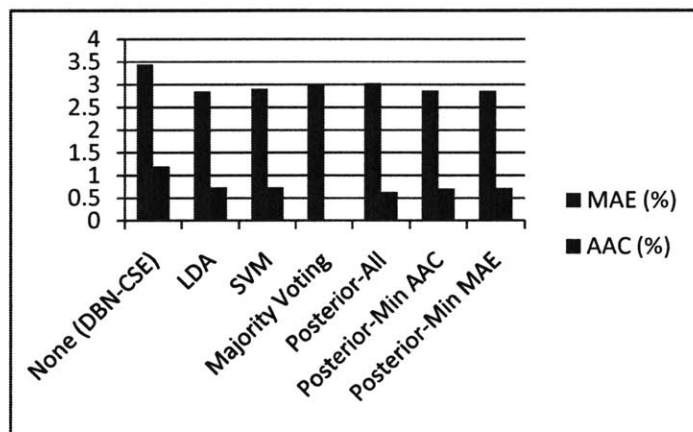
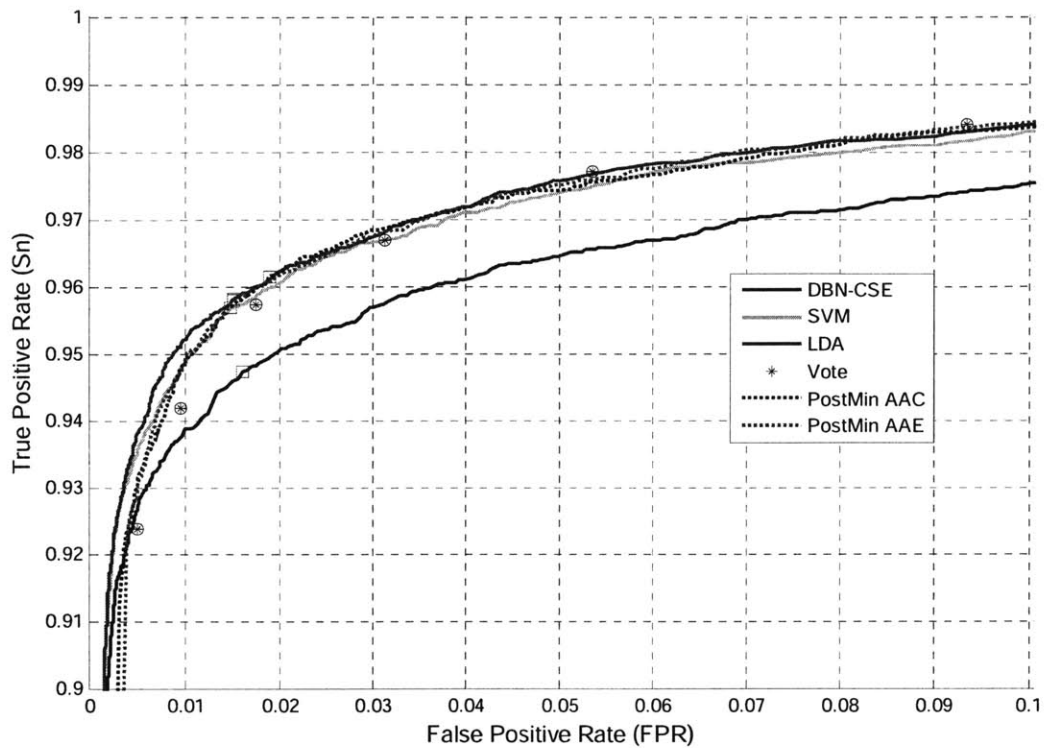


Figure 5.1: ROC curves and Performance of different methods of metric combinations.

6 *de novo* Gene Prediction with Conditional Random Fields

A common concern with a classification approach to gene prediction has been that, no matter how good the discriminative metric, in the *de novo* gene prediction problem, segmented sequences are not available. While this is certainly true with the generative models that have been the norm in computational gene prediction, it is not the case with discriminative models such as linear chain semi-CRFs. Another reason for the concern is that even though a good discriminative metric may be able to discriminate between exons and non-coding sequences, its performance is uncertain on sequences that are partly coding or partly exons. Throughout this work, I have maintained the claim that a discriminative metric possesses the desirable qualities of a CRF feature function. In this section, I test this claim by incorporating the Dynamic Bayesian Network of Codon Sequence and Evolution (DBN-CSE) Metric, the best discriminative metric, into a Conditional Random Field framework to build a *de novo* gene predictor as shown in Figure 6.1. A maximum entropy splice site scoring feature [36] is used in conjunction with the DBN-CSE metric to score the exon-intron junction potentials. The overall segment transition diagram for the model is also shown in Figure 6.1. The exon label E_i^+ denotes an exon segment on the forward coding strand with i nucleotides belonging to an incomplete codon at boundary corresponding to the 3' end of the forward strand. Therefore, only exons with 0 incomplete codons can be followed by a segment labeled "Intergenic (IG)". Otherwise, exons can only be followed by introns with the same incomplete codon memory label. Segmentation with such a model is performed from the 5' to 3' direction of the forward strand.

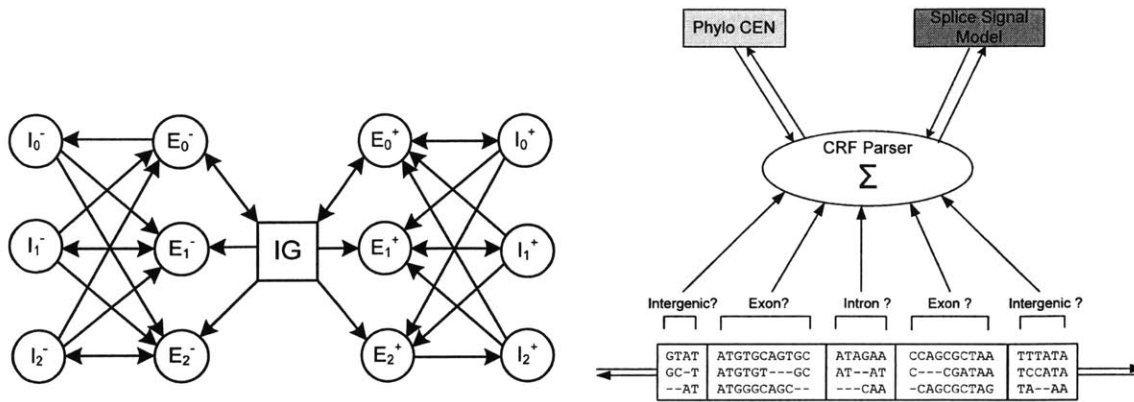


Figure 6.1: Segment transition diagram and architecture of the semi-CRF gene predictor.

The CRF gene predictor was used to annotate randomly chosen regions of the *Drosophila melanogaster* genome and the resulting segmentation was compared with the actual annotations of genes in those regions.

To evaluate the CRF gene parser, a set of 40 genes with 120 exons was chosen randomly from the genome of *D. melanogaster*. The genes were not selectively chosen to be reliably annotated or well conserved. Intergenic sequences up to, in some cases, 10000 nucleotides flanking the genes were included to create 10 sequences containing multiple genes on both forward and reverse strands. The CRF gene parser was then used to perform segmentation on these sequences to annotate the genes. The results were compared with the FlyBase annotations and are presented in Table 6.1. The performance of the gene predictor was evaluated in terms of nucleotide sensitivity and precision as well as exon sensitivity and precision. An exon was considered correct only if the predicted exon overlapped exactly with the annotation, with no boundary errors.

Table 6.1 The raw annotation scores of the CRF gene model parser on the unrestricted genome set

Seq ID	Seq. Length	Number of Exons	Annotated		Predicted			
			Length of Coding Regions	Length of Non-Coding Regions	Nucleotides Correctly Annotated	Coding Nucleotides Reported	Exons Annotated Exactly Correct	Exons Reported
1	3700	7	2712	988	2712	2712	7	7
2	3850	7	2082	1768	1848	1881	5	7
3	3000	6	1407	1593	1407	1422	6	5
4	5000	10	2658	2342	2658	2718	10	9
5	20100	16	5889	14211	5858	7452	15	27
6	13000	12	7371	5629	7128	7296	5	12
7	11000	11	6663	4337	6354	6552	7	12
8	17000	4	786	16214	317	339	2	2
9	22000	22	10527	11473	10481	11163	18	27
10	21600	25	11214	10386	11169	11295	23	26
Total	120250	120	51309	68941	49932	52830	98	134

The CRF was found to achieve *de novo* gene segmentation at a Nucleotide Sensitivity and Precision of 97% and 95% respectively with Exon Sensitivity and Precision of 82% and 77% respectively. The performance of the semi-CRF gene model with a single discriminative metric feature and no additional training is found to rival the performance of N-SCAN [3], the current state of the art large scale gene predictor on gene prediction in the Fruit Fly, as shown in Figure 6.2.

Even with the accuracy being comparable to that of the state of the art, the CRF gene prediction framework offers a number of advantages. Firstly, the CRF contains far fewer parameters than the N-SCAN architecture. Secondly, all of the features are trained independently, and bottom up, allowing full transparency at all levels. Finally, the CRF architecture permits a large amount of evolvability and flexibility in the addition of new features that could potentially detect other significant features in the genome such as CPG islands, motifs and other signals important in the regulation of genes.

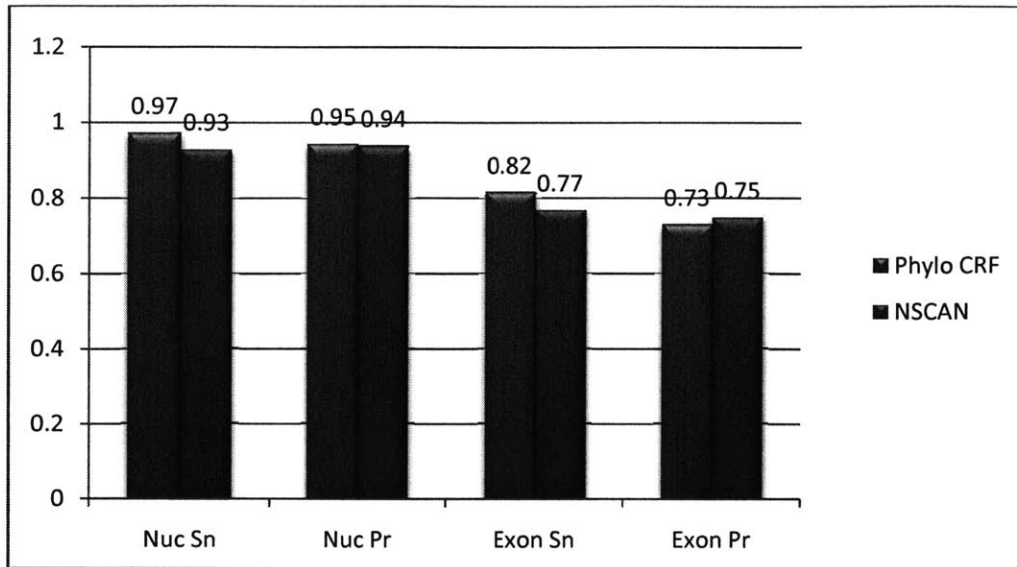


Figure 6.2: The *de novo* gene prediction accuracy of the semi-CRF gene predictor compared to that of NSCAN.

7 Future Work

The Conditional Random Field gene predictor with the novel phylogenetic metric feature (presented in Chapter 6) was shown to be successful at *de novo* (un-segmented) gene annotation, over a randomly chosen subset of the *D. melanogaster* genome, with a nucleotide accuracy of 95%. Motivated by these results, I plan to increase the sophistication of the model by incorporating more features and finally evaluating the entire genome of the Fruit Fly. Once the system is shown to be reliable at gene prediction by integrating multiple lines of evidence through the use of multiple features, I plan on evaluating and annotating the entire Human genome, with the help of whole genome alignments of 24 mammals that are currently being sequenced. In this process, I also hope to discover new biology, through the analysis of the performance of these models.

Furthermore, as a step towards my PhD thesis, I plan to investigate a number of directions for improving these models which tackle problems not addressed by current gene predictors. For example, in larger organisms, especially humans, there are a significant number of alternatively spliced genes which can create a number of different protein products by splicing in and out certain exons. Because current gene predictors only produce a single maximum likelihood annotation of a gene, which is often only one of many potential transcripts, alternatively spliced genes can only be detected through wet-lab experiments. I plan to generalize my system to Bayesian models, such as the recently proposed Bayesian Conditional Random Fields [37] that could potentially produce a posterior probability on a number of different annotations of the same gene, thereby discovering these alternatively-spliced transcripts computationally.

8 Contributions

With the unprecedented number and size of genomes currently being sequenced, including 24 mammals related to human [38], there is an urgent need for reliable computational gene predictors that can leverage the observed evolution of aligned sequences to improve gene annotation accuracy in these genomes, a task difficult for current generative gene models [3]. The recent emergence of discriminative models such as Conditional Random Fields (CRFs) show promise, but lack the discriminative feature functions required to build accurate phylogenetic gene models. In this thesis, I presented a number of novel phylogenetic discriminative metrics that were shown to possess all the desirable characteristics of such feature functions, and were shown to perform *de novo* gene prediction with 95% nucleotide accuracy on alignments of the Fruit Fly.

My contributions are detailed in the following sections.

8.1 Novel Metrics for Exon Classification

I presented a number of novel metrics that, individually, were shown to achieve very high accuracy in the classification of 50,000 protein-coding and non-coding sequence alignments from the Fruit Fly genome, outperforming all other existing metrics. The DiCodon and Conservation Periodicity (sequence-based) metrics classified exons with an accuracy of over 95%, a relative reduction in error of 28% over the best performing sequence-based existing metric. The Phylogenetic Codon Evolution Network and Dynamic Bayesian Network of Codon Sequence and Evolution, (evolutionary metrics) modeled the observed evolution of codon sequences in a phylogenetic framework, to achieve an exon classification accuracy of over 96%, a relative reduction in error of 20% over currently existing evolutionary metrics. Furthermore, I demonstrated that these novel metrics steadily improved in classification performance as the number of species in the alignment was increased.

8.2 Posterior Framework for Metric Combinations

I presented a number of methods that enabled the combination of an arbitrary set of metrics in a unified probabilistic framework. Each of the four methods of metric combination were shown to improve the overall classification performance and in some cases further reducing the error over the best single classifier by 17% relative. The combination framework was also shown to be robust to posterior density estimation errors.

8.3 Integrating Metrics into a Conditional Random Field Gene Model

Finally, I incorporated the novel Dynamic Bayesian Codon Evolution Network metric into a discriminative conditional random field gene model. I evaluated the performance of the *de novo* gene predictor on unsegmented genome alignments of the Fruit Fly, and achieved state of the art performance of 97% nucleotide sensitivity and 95% nucleotide precision.

9 References

1. International Human Genome Sequencing Consortium, *Finishing the euchromatic sequence of the human genome*. Nature, 2004. **431**: p. 931-945.
2. Gregory, T.R., *The Evolution of the Genome*. 2005, London: Elsevier.
3. Gross, S.S., Brent, Michael, R., *Using Multiple Alignments to Improve Gene Prediction*. Journal of Computational Biology 2006. **13**(2): p. 379-393.
4. Siepel, A.C., Hausler, D., *Computational Identification of Evolutionarily Conserved Exons*, in *Proceedings of the Eighth Annual International Conference on Computational Molecular Biology* 2004. p. 177-186.
5. Vinson, J., et al. *Comparative Gene Prediction using Conditional Random Fields*. in *Advances in Neural Information Processing Systems 19*. 2006.
6. Sutton, C. and A. McCallum, *An Introduction to Conditional Random Fields for Relational Learning*, in *Introduction to Statistical Relational Learning*, L. Getoor and B. Taskar, Editors. 2006, MIT Press, To Appear.
7. Fly Consortium, *Sequencing of 12 Drosophila Genomes*. In Preparation, 2007.
8. Rabiner, L., *A tutorial on hidden Markov models and selected applications in speech recognition*, in *Readings in speech recognition*. 1990, Morgan Kaufmann Publishers Inc. p. 267-296.
9. Dietterich, T., *Machine Learning for Sequential Data: A Review*, in *Lecture Notes in Computer Science*, T. Caelli, Editor. 2002, Springer-Verlag.
10. Burge, C. and S. Karlin, *Prediction of complete gene structures in human genomic DNA*. J Mol Biol, 1997. **268**(1): p. 78-94.
11. Murphy, K., *Dynamic Bayesian Networks: Representation, Inference and Learning*, in *Computer Science Division*. 2002, UC Berkeley.

12. Felsenstein, J., *Evolutionary trees from DNA sequences: a maximum likelihood approach*. J. Mol. Evol., 1981. **17**: p. 368-376.
13. Durbin, R., et al., *Biological Sequence Analysis: Probabilistic Models of Proteins and Nucleic Acids*. 1998, Cambridge: Cambridge University Press.
14. Korf, I., et al., *Integrating Genomic Homology Into Gene Structure Prediction*. Bioinformatics, 2001. **17**(1): p. S140-S148.
15. Siepel, A. and D. Haussler, *Combining phylogenetic and hidden Markov models in biosequence analysis*. J Comput Biol, 2004. **11**(2-3): p. 413–428.
16. Sarawagi, S. and W. Cohen. *Semi-Markov Conditional Random Fields for Information Extraction*. in *Proceedings of ICML*. 2004.
17. Sarawagi, S., *Efficient inference on sequence segmentation models*, in *Proceedings of the 23rd international conference on Machine learning*. 2006, ACM Press: Pittsburgh, Pennsylvania.
18. Lafferty, J., A. McCallum, and F. Pereira. *Conditional random fields: Probabilistic models for segmenting and labeling sequence data*. in *Proc. 18th International Conf. on Machine Learning*. 2001.
19. Wallach, H.M., *Conditional Random Fields: An Introduction*, in *Technical Report MS-CIS-04-21*. 2004, Dept. of Computer and Information Science, University of Pennsylvania.
20. Anastassiou, D., *Genomic Signal Processing*, in *IEEE Signal Processing Magazine*. 2001. p. 8-20.
21. Sussillo, D., A. Kundaje, and D. Anastassiou, *Spectrogram Analysis of Genomes*. EURASIP Journal on Applied Signal Processing, 2004. **2004**(1): p. 29-42.
22. Tiwari, S., et al., *Prediction of probable genes by Fourier analysis of genomic sequences*. Computer Applications in the Biosciences, 1997. **13**(3): p. 263-270.
23. Guo, F.B., H.Y. Ou, and C.T. Zhang, *ZCURVE: A New System For Recognizing Protein-Coding Genes in Bacterial and Archaeal Genomes*. Nucleic Acids Research, 2003. **31**: p. 1780-1789.

24. Gao, F., Zhang, C.-T., *Comparison of Various Algorithms for Recognizing Short Coding Sequences of Human Genes*. *Bioinformatics*, 2004. **20**(5): p. 673-681.
25. Salzberg, S.L., et al., *Microbial Gene Identification Using Interpolated Markov Models* *Nucleic Acids Research* 1998. **26**(21998).
26. Delcher, A., Harmon, D., Kasif, S., White, O., Salzberg, S., *Improved Microbial Gene Identification with GLIMMER*. *Nucleic Acids Research*, 1999. **27**(23): p. 4636-4641.
27. Kellis, M., et al., *Sequencing and comparison of yeast species to identify genes and regulatory elements*. *Nature*, 2003. **423**(6937): p. 241-254.
28. Lin, M., *Comparative Gene Identification in Mammalian, Fly, and Fungal Genomes* in *Computer Science*. 2006, MIT.
29. Henikoff, S. and J.G. Henikoff, *Amino Acid Substitution Matrices from Protein Blocks*. *PNAS*, 1992. **89**(22): p. 10915-10919.
30. Duda, R.O., P.E. Hart, and D.G. Stork, *Pattern Classification*. 2000: Wiley-Interscience.
31. Blanchette, M., et al., *Aligning multiple genomic sequences with the threaded blockset aligner*. *Genome Research*, 2004. **14**: p. 708-715.
32. Drysdale, R., *The Drosophila melanogaster genome sequencing and annotation projects: A status report*. *Briefings in Functional Genomics and Proteomics*, 2003. **2**(2): p. 128-134.
33. Reiter, L.T., et al., *A Systematic Analysis of Human Disease-Associated Gene Sequences In Drosophila melanogaster*. *Genome Research*, 2001. **11**(6): p. 1114-1125.
34. Powell, A., *Progress and Prospects in Evolutionary Biology: The Drosophila Model*. 1997, New York: Oxford University Press, Inc.
35. Joachims, T., *Learning to Classify Text Using Support Vector Machines*. 2002: Kluwer.
36. Yeo, G., Burge, C.B., *Maximum Entropy Modeling of Short Sequence Motifs with Applications to RNA Splicing Signals*. *Journal of Computational Biology*, 2004. **11**: p. 377-394.

37. Qi, Y., M. Szummer, and T.P. Minka. *Bayesian Conditional Random Fields*. in *Proceedings of AISTATS 2005*.
38. Cooper, G.M., et al., *Distribution and intensity of constraint in mammalian genomic sequence*. *Genome Research*, 2005. **15**: p. 901-913.

Appendix A:

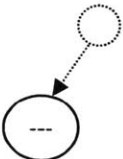
The Algorithm PBNJ(T,C,X) is a recursive algorithm that calculates the Phylogenetic Bayesian Network Joint Probability for a tree T with observed codons at the leaves, a set C of Conditional Probability Tables along each branch evaluated at node x. The initial call is made with x = Root of the tree.

$p = \text{PBNJ}(T, C, x)$

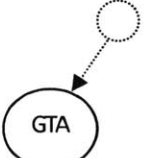
```

switch [T.left(x) T.right(x) T.val(x)]
  case [0 0 0]
    return [1 1 1 ... 1]T
  case [0 0 1]
    return [0 0 ... 0 1 0 ... 0]T
  case [1 1 0]
    return (C{x,left}*PBNJ(T, C, T.left(x))) .*
           (C{x,right}*PBNJ(T, C, T.right(x)))
  case [1 0 1]
    return C{x,left}(T.val(x), :)*PBNJ(T, C, T.left(x))
  case [1 0 0]
    return C{x,left}*PBNJ(T, C, T.left(x))
  case [0 1 0]
    return C{x,right}*PBNJ(T, C, T.right(x))

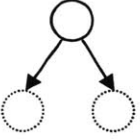
```



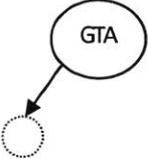
Missing Informant




Observed Informant



Ancestral node with two children



Root Node

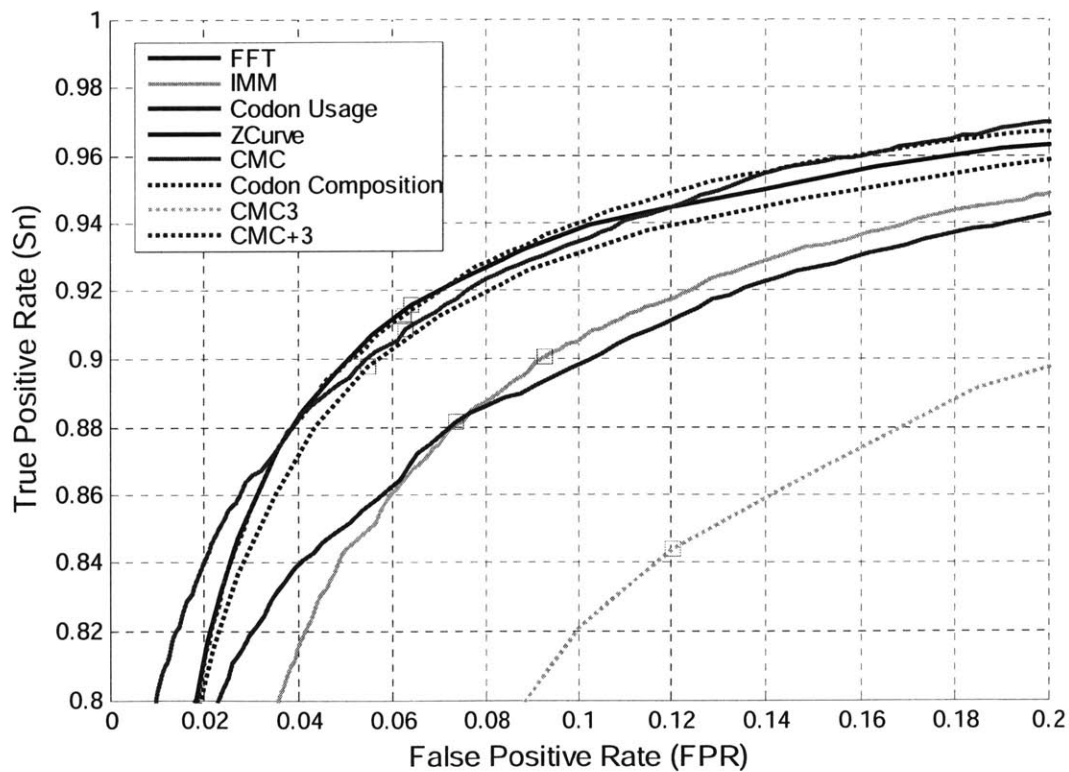


Ancestral node with left child

Appendix B:

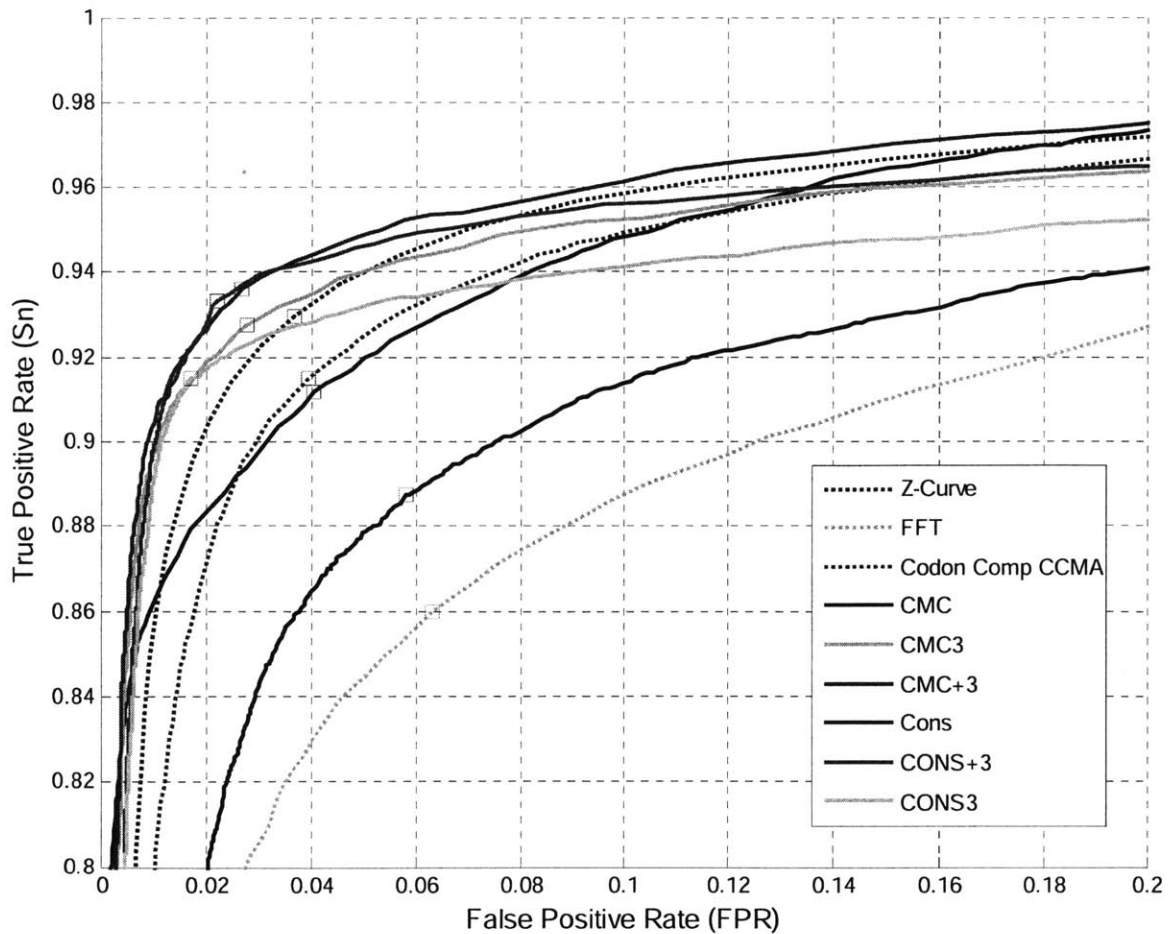
B1. The comparative analysis of classification performance of single sequence metrics

Metric	Abbreviation	Minimum Average Error			Area Above Curve	Parameters
		Sn (%)	Fpr (%)	MAE (%)	AAC (%)	
Three-Base Periodicity	fft	72.89	13.32	20.22	14.16	0
Interpolated Markov Model	imm	90.07	9.30	9.62	4.42	24576
Codon Usage	cus	88.16	7.38	9.61	3.93	82
Z-Curve Score	zcs	91.62	6.41	7.40	3.10	69
Codon Composition	ccs	89.80	5.48	7.84	3.36	63
Codon Markov Chain	cmc	90.89	6.29	7.70	2.26	8190
DiCodon Periodicity	cmc3	84.40	12.03	13.82	8.33	8190
Codon Markov Chain with Periodicity	cmc+3	91.28	6.23	7.48	2.99	8190

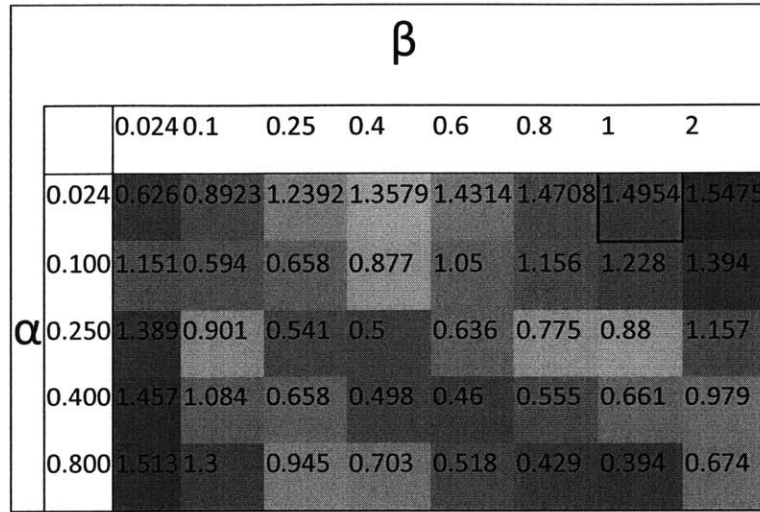


B2: Comparative analysis of the sequence-based metrics on the classification of alignments of exons.

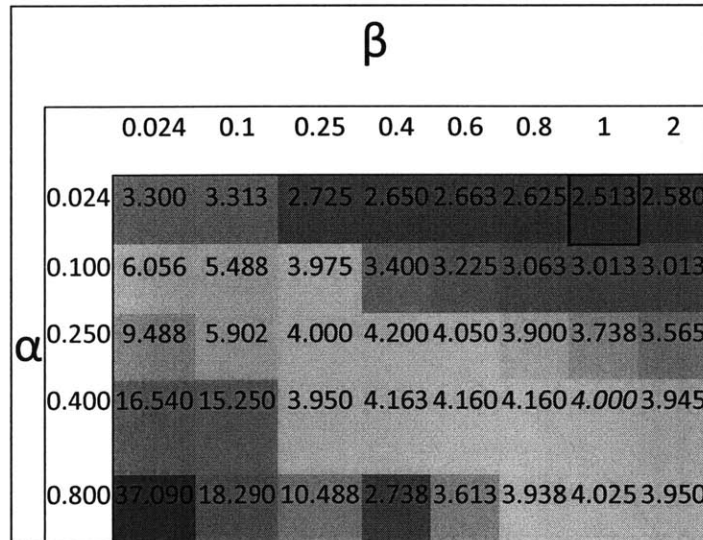
Metric	Abbreviation	Minimum Average Error			Area Above Curve	Parameters
		Sn (%)	Fpr (%)	MAE (%)	AAC (%)	
3-Base Periodicity Alignment	FFT					
Codon Markov Chain	CMCA	91.16	4.01	6.43	1.84	8190
DiCodon Periodicity	CMCA3	92.77	2.77	5.00	2.57	8190
CMC + DiCodon Periodicity	CMCA+3	93.61	2.65	4.52	1.59	12285
Z Curve Score Alignment	ZCS	91.48	3.91	6.22	2.76	69
Codon Composition Alignment	CCMA	92.95	3.65	5.35	2.26	63
Conservation	CONS	88.72	5.79	8.54	4.63	0
Conservation Periodicity	CONS3	91.49	1.70	5.11	3.45	0
Conservation + Periodicity	CONS+3	93.31	2.19	4.44	2.50	0



B3. Rate of Evolution Parameter Estimation for the Phylogenetic Codon Evolution Network.



Model Likelihood Error (Generative Parameter Estimation)

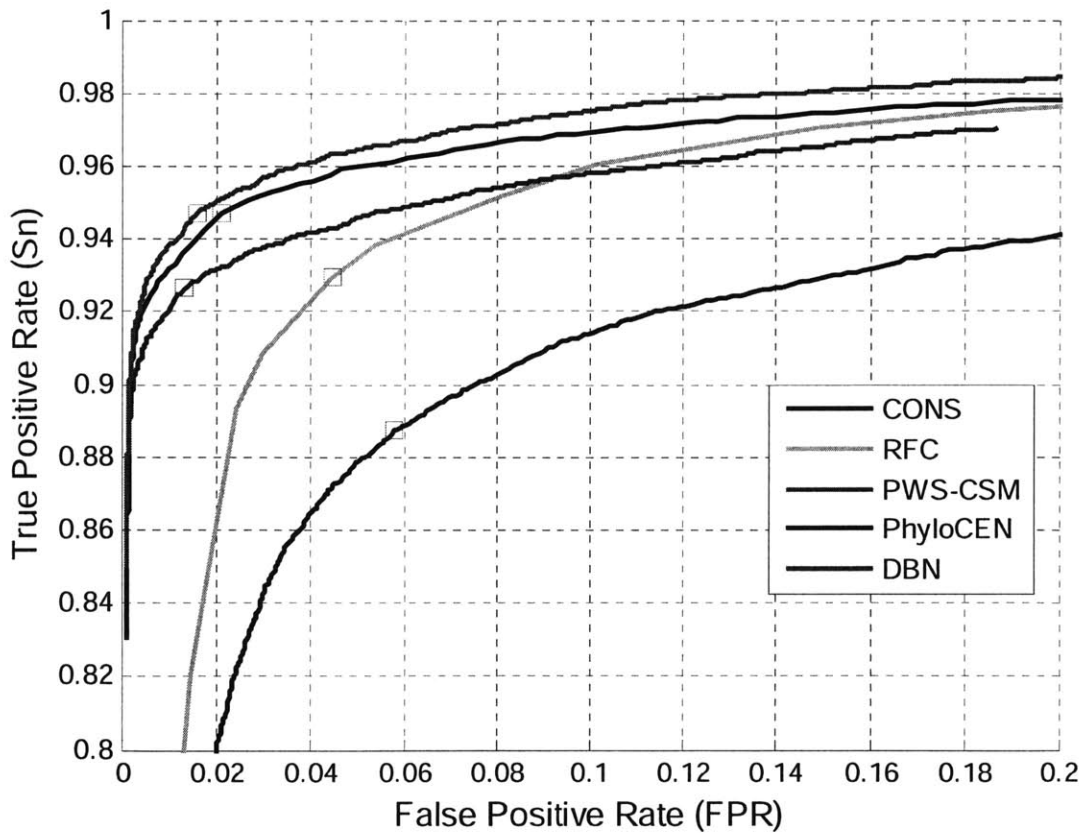


Minimum Average Error Rate Classification Error (Discriminative Parameter Estimation)

Figure B: The variation of the generative and discriminative error objective functions for different values of evolution rate parameters α and β .

B4. Comparative analysis of the evolutionary-signature metrics on the classification of alignments of exons.

Metric	Abbreviation	Minimum Average Error			Area Above Curve	Parameters
		Sn (%)	Fpr (%)	MAE (%)	AAC (%)	
Reading Frame Conservation	RFC	92.95	4.46	5.76	2.48	0.00
Conservation	CONS	88.72	5.79	8.54	4.63	0.00
Conservation and Periodicity	CONS+3	93.31	2.19	4.44	2.50	0.00
CSM Pairwise Summed	PW-CSM	92.66	1.33	4.34	2.16	88704
PhyloCEN	PhyloCEN	94.69	2.11	3.71	1.43	8066
DBN	DBN	94.72	1.61	3.45	1.20	16256



Appendix C:

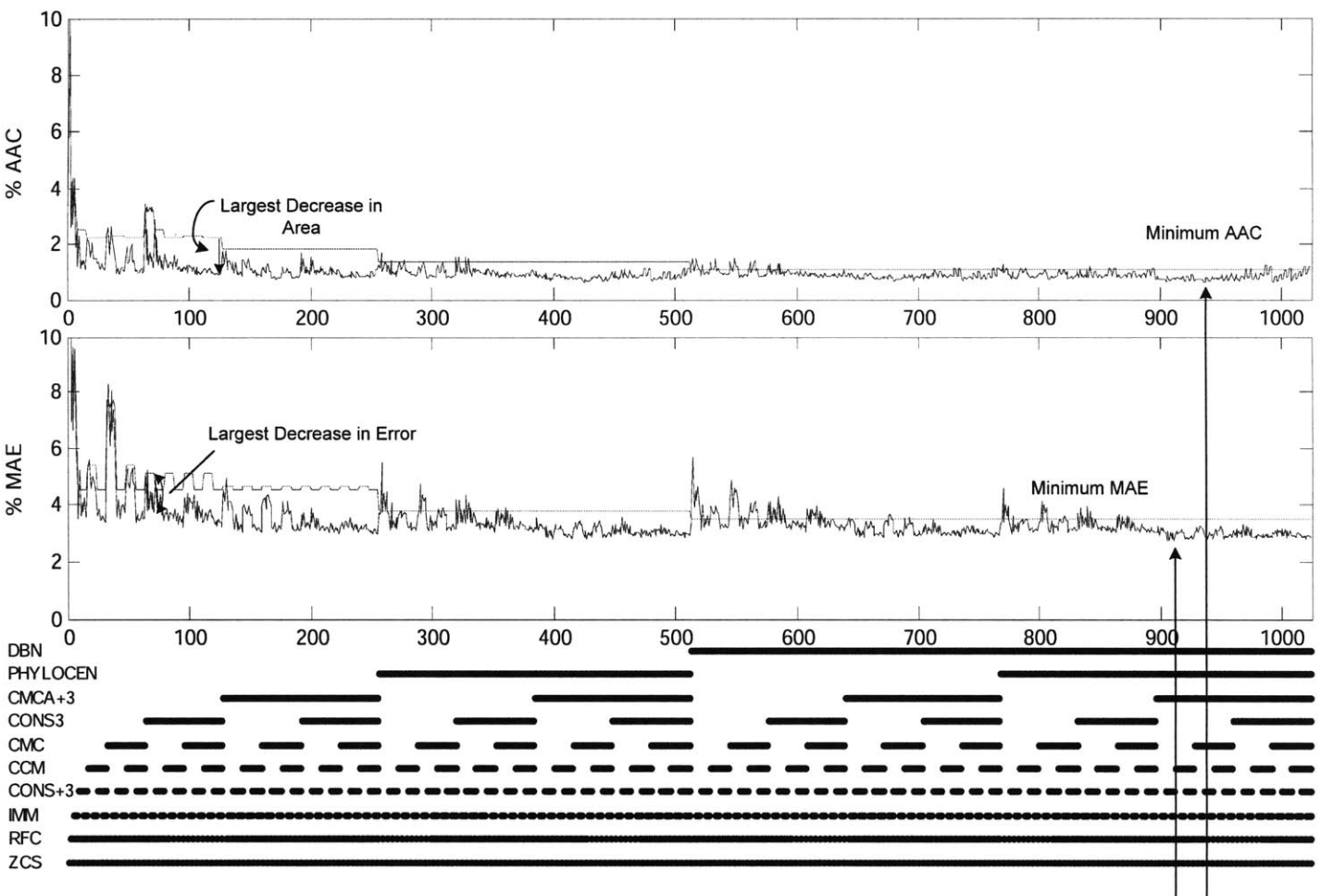


Figure C: The Average Error (MAE) and Area above curve (AAC) for every combination of metric (shown in blue) as compared to the MAE and AAC for the best performing single metric in that combination.

6195-24