# Genomic Data Mining for the Computational Prediction of Small Non-coding RNA Genes

A Thesis
Presented to
The Academic Faculty

by

Thao Thanh T. Tran

In Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy in the
School of Electrical and Computer Engineering

Georgia Institute of Technology
May 2009

**Genomic Data Mining for the Computational Prediction of Small Non-coding RNA Genes**

Approved by:


Professor Xiaoli Ma, Committee Chair
School of Electrical and Computer Engineering
*Georgia Institute of Technology*

Professor G. Tong Zhou, Advisor
School of Electrical and Computer Engineering
*Georgia Institute of Technology*

Professor Ying Xu
Department of Biochemistry and Molecular Biology
*University of Georgia*

Professor Arthur Koblasz
School of Electrical and Computer Engineering
*Georgia Institute of Technology*

Professor Eberhard Voit
Department of Biomedical Engineering
*Georgia Institute of Technology*



Date Approved: 12 January 2009

*To my family and Andrew for all your encouragement and love.*

# ACKNOWLEDGEMENTS

I have been blessed with the great privilege to work with the best minds during my dissertation work and would like to express my sincere appreciation towards everyone that has helped me along the way.

First, I would like to express my sincere gratitude towards my advisors, Dr. Ying Xu and Dr. G. Tong Zhou, for supporting me and giving me the opportunity to pursue my Ph.D. degree. Dr. Zhou has been a great inspiration in introducing me to the field of genomic signal processing and in guiding me through all the challenges of this field. She is a wonderful mentor and I am very fortunate to have her encouragement throughout the years. I would like to express my great appreciation to Dr. Xu who has constantly supported my ideas and work. He has always made time to support and guide me throughout my research pursuits. I have learned so much about what it takes to be a great researcher from him. His work ethics, motivation, and kindness are the qualities that I hope to strive for in my career. I feel honored to have had the opportunity to work with him.

Additionally, I would like to thank my dissertation committee members: Dr. Xiaoli Ma, Dr. Arthur Koblasz, and Dr. Eberhard Voit for providing guidance and suggestions to improve my thesis. I am grateful for my collaboration with members of the Computational Systems Biology Lab (CSBL) at the University of Georgia (in alphabetical order): Gregory Baramidze, Dongsheng Che, Huiling Chen, Jacky Chou, Juan Cui, Phuongan Dam, Kyle Ellrott, Claire Gervais, Juntao Guo, Dorothy Hammond, Chindo Hicks, Guojun Li, Bingqiang Liu, Huiqing Liu, Qi Liu, Jizhu Lu, Fenglou Mao, Victor Olman, Yongbin Ou, Zhengchang Su, Hongwei Wu, Kun Xu, Yanbin Yin, Chan Zhou, Fengfeng Zhou, Wen Zhou, and others. The synergistic environment and open collaboration atmosphere at CSBL is truly exceptional.

I would also like to thank my Georgia Tech group members: Vince Emanuele, Chunming Zhao, Bob Baxley, Kun Shi, Ning Chen, Raviv Raich, Lei Ding, Hua Qian, and

Chunpeng Xiao for their insightful discussions. I thank Vince Emanuele for his advice and encouragement throughout the course of my research.

Last but not least, I dedicate my work to the strong unconditional support of my mom and dad for their sacrifices to enable my siblings and I to be where we are today. Many thanks to my older sister for reminding me how proud she is of me, my older brother for never doubting me, and my little sister for sharing the ups and downs of research life. I also dedicate my work to Andrew, the most important man in my life, for his timeless support, enduring patience, devoted kindness, endless encouragement, and unconditional love. There are not enough words to describe how grateful I am to have Andrew and my family with me through this journey.

# TABLE OF CONTENTS

# LIST OF TABLES

# LIST OF FIGURES

# SUMMARY

The objective of this research is to develop a novel computational prediction algorithm for non-coding RNA (ncRNA) genes using features computable for any genomic sequence without the need for comparative analysis. Existing comparative-based methods require the knowledge of closely related organisms in order to search for sequence and structural similarities. This approach imposes constraints on the type of ncRNAs, the organism, and the regions where the ncRNAs can be found. We have developed a novel approach for ncRNA gene prediction without the limitations of current comparative-based methods. Our work has established a ncRNA database required for subsequent feature and genomic analysis. Furthermore, we have identified significant features from folding-, structural-, and ensemble-based statistics for use in ncRNA prediction. We have also examined higher-order gene structures, namely operons, to discover potential insights into how ncRNAs are transcribed. Being able to automatically identify ncRNAs on a genome-wide scale is immensely powerful for incorporating it into a pipeline for large-scale genome annotation. This work will contribute to a more comprehensive annotation of ncRNA genes in microbial genomes to meet the demands of functional and regulatory genomic studies.

# CHAPTER I

# INTRODUCTION

The objective of this research is to develop a novel computational prediction method for non-coding RNA (ncRNA) genes. Non-coding RNA genes function without being translated into proteins. The majority of annotation analyses in the past two decades have focused on identifying protein-coding genes. While important, the identification of ncRNA genes did not receive much attention until its recent discovery in the regulation of a diverse range of cellular processes. Previous approaches to ncRNA prediction primarily relied on homology-based methods to search for sequence and structural similarities across different genomes. While most existing methods attempt to find conserved functional ncRNAs, there is an urgent demand for new algorithms that do not rely on the knowledge of closely related organisms. This approach will be advantageous, as more diverse genomes are being sequenced each year. This work will contribute to a more comprehensive annotation of ncRNA genes in microbial genomes to meet functional and regulatory genomic studies. To address this challenging issue, we propose to develop a feature-based learning method to predict ncRNA genes.

The main contributions of this dissertation work address a challenging problem in the interdisciplinary field of bioinformatics by proposing a novel algorithm to identify microbial ncRNA genes on a genome-wide scale. A further contribution is the additional biological insight gained from the features used to distinguish between the ncRNA genes and genomic background. The broader contribution of the computational prediction tool is its potential use in existing pipelines to perform routine annotation of ncRNA genes, similar to how protein coding genes are currently annotated. The annotation of ncRNA genes paves the way for understanding its role in cellular regulation and complex diseases.

## *1.1 Background*

In recent years, there has been enormous interest in the field of signal processing to apply techniques from pattern recognition, controls, and dynamic modeling to the growing interdisciplinary field of bioinformatics [5, 36, 37, 54, 74, 89, 116, 139, 148, 138, 170]. Signal processing offers valuable tools to tackle new and challenging problems in bioinformatics. In this work, we propose a data mining-based approach for the genomic annotation of non-coding RNA genes. We first give a brief review of the central dogma of molecular biology. Next, we provide the motivation for our problem by identifying the existing challenges and approaches in non-coding RNA gene finding.

### 1.1.1 The need for genome annotation

The completed draft of the Human Genome Project (HGP) in 2001 [86, 149] represents a landmark milestone to an 11-year international collaborative project to decipher the three billion nucleotides that code for life. In the mid-1990s, there were fewer than half a dozen published complete genomes [92]. With the draft of the HGP, the number of published genomes grew to 72. Today, there are more than 481 published genomes and more than 100 more unpublished genomes. With more than 170 Gigabases of nucleotides and almost 2,000 more ongoing sequencing projects, researchers in the field of bioinformatics are faced with the daunting task of how to interpret this data. The demand for accurate labeling of this data is at the focus of genome annotation. With reliable annotation, researchers have the blueprint by which they can explore the complexity of life.

### 1.1.2 Central dogma

The code for all life and its complexity is encoded in DNA sequences consisting of nucleotide (nt) bases (`A`, `C`, `G`, `T`). Within the DNA sequence, there are regions called genes that encode specific functions. These genes form the basis for all annotation work in an organism. The central dogma of molecular biology is the information transfer from DNA to RNA to protein, as summarized in Figure 1.1. DNA is converted to messenger RNA (mRNA) through the process of transcription in which nucleotide `T` is replaced by nucleotide `U`. Through the


**Figure 1.1:** Central dogma of molecular biology.

process of translation, sequences of three nucleotides (codons) on the mRNA, representing the code for a specific amino acid, are assembled into proteins with the help of other RNAs. These proteins help carry out many cellular processes and were once thought to be the only functional biomolecule in a cell.

### 1.1.3 Review of non-coding RNA genes

The majority of annotation analyses in the past two decades have focused on identifying protein-coding genes [49, 50, 125]. As such, the other class of genes that function as RNA without being translated into protein, called non-coding RNA (ncRNA) genes or small non-coding RNA (sRNAs) genes, have mostly been overlooked as intermediary molecules that help translate mRNA into protein. It was not until Fire and Mello's 1998 Nobel-winning discovery of how some small RNAs could switch off certain mRNAs (RNA-interference) [51] that researchers began to change their view on RNA. In humans, it is estimated that about 98% of the genome gets transcribed, of which only 2% correspond to protein coding genes [102, 103, 140]. Furthermore, the complexity of an organism is not proportional to the number of protein coding genes. Besides mechanisms like alternative splicing, regulation by ncRNA genes is believed to account for an organism's complexity [102, 103]. The identification of ncRNAs is needed for a better understanding of the entire genomic landscape of an organism. It is hoped that the knowledge gained from ncRNAs will play a major role in shaping our view of diseases in the coming century. Mutations in ncRNAs and their associated proteins have been implicated in genetic diseases, neurological disorders, and cancer [39, 67, 57, 112, 17]. The identification and better understanding of these ncRNAs are crucial for the development of effective therapies in these and other viral-based diseases such as HIV, hepatitis, influenza, and SARS [57].

Non-coding RNAs are involved in various aspects of cellular processes, including regulation of gene expression, controlling activation of regions in chromosomes, intron excision, DNA packaging, and RNA modification and editing [58, 66]. The major classes of RNA

as modified from [16] are shown in Figure 1.2. Within the ncRNAs, transfer RNA (tRNA) and ribosomal RNA (rRNA) have important roles in protein synthesis. Current computational methods take advantage of the structural and sequence conservation of tRNA [98] and rRNA [84] to identify them across species. As such, these two particular classes of ncRNA are not included in the scope of the proposed work. The class of interest in our work involves ncRNAs ranging in size from about 20-1000 nt. The various ncRNA classes as modified from [13, 101] are summarized in Table 1.1.

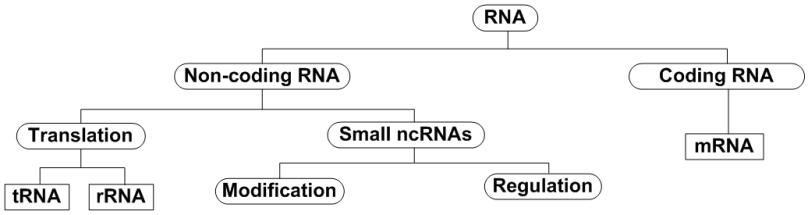


**Figure 1.2:** Major classes of RNA.

**Table 1.1:** Classes of small non-coding RNAs with the function, approximate size, organism where found, and references for existing computational prediction methods. The abbreviations represent (in order of appearance): snRNA, small nuclear RNA; snoRNA, small nucleolar RNA; gRNA, guide RNA; SRP, signal recognition particle; miRNA, micro RNA; siRNA, small interfering RNA; piRNA, piwi-interacting RNA; rasiRNA, repeat associated siRNA; tmRNA, transfer messenger RNA.

| Class | Function | Size (nt) | Organism | Reference |
|---|---|---|---|---|
| RNA processing and modification | | | | |
| RNase P RNA | tRNA/rRNA maturation | 220-440 | All | [61] |
| snRNA | mRNA splicing | 100-160 | Eukarya | |
| C/D snoRNA | tRNA/snRNA/rRNA methylation | 60-80 | Archaea, Eukarya | [99] |
| H/ACA snoRNA | snRNA, rRNA pseudouridylation | 130 | Eukarya | [134, 133, 43] |
| gRNA | RNA editing | 40-80 | Kinetoplastids | |
| 4.5S RNA SRP | protein secretion | 300-400 | Eukarya | |
| Regulation | | | | |
| miRNA | gene silencing | 21-24 | Multicellular | [90, 91] |
| siRNA | gene silencing | 22 | Multicellular | |
| piRNA | gene silencing | 26-32 | Mammals | [19] |
| rasiRNA | gene silencing | 24-29 | Drosophila | |
| antisense RNA | gene silencing/activation | 50-250 | Bacteria | [58] |
| 6S RNA | transcription regulation | 200 | Bacteria | |
| Genome stability | | | | |
| telomerase RNA | telomerase synthesis | 150-1300 | Eukarya | |
| Translation | | | | |
| tmRNA | releases stalled mRNA | 300-400 | Bacteria | [88] |

Because of the laborious nature of wet lab identification of ncRNAs [67], it is necessary to

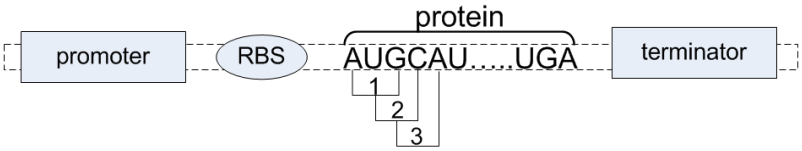


**Figure 1.3:** Sequence signals for protein-coding genes. Protein coding genes are characterized by conserved sequence motifs in the promoter, ribosomal binding site (RBS), and terminator regions. Additionally, proteins contain open reading frames as shown by frames 1-3 that code for specific start and stop codons.

find computational methods to narrow the search space of potential candidates. Given the importance of identifying ncRNAs, we next explore, in the following sections, the challenges in predicting ncRNAs and the limitations of current approaches.

### 1.1.4 Challenges in computational non-coding RNA gene finding

The identification of ncRNA genes is more challenging than protein coding genes because of the lack of sequence signals. Unlike protein coding genes, non-coding RNA genes do not contain signals such as open reading frames (i.e., sequence containing a start through a stop codon), codon bias (hexamer frequency), or ribosome binding sites (RBS), as shown in Figure 1.3. Although some ncRNA genes have signals such as promoters and terminators [6, 21], the identification of these transcriptional features is not easily recognizable and also varies depending on which RNA polymerase is used for transcription [40]. Furthermore from Table 1.1, several classes of ncRNA genes have much shorter length as compared to protein coding genes, which tend to be longer than 900 nt [15]. The smaller size makes it more challenging to find ncRNAs using similarity searches as commonly used in protein gene finding [4].

### 1.1.5 Review of computational methods for non-coding RNA gene finding

Methods for ncRNA gene finding can be roughly categorized into two classes: those that identify members based on prior knowledge of a family of ncRNAs and those that find new ncRNAs [105, 13, 39]. We briefly summarize both approaches and discuss in more detail the latter since finding novel ncRNAs is the main focus of the proposed work. We exclude discussion of miRNA prediction since this class of ncRNAs is not currently known

to exist in prokaryotic organisms and very specific algorithms already exist to identify them [90, 85, 14, 62, 76, 155, 164, 171].

#### *1.1.5.1 Methods to detect new members of well-characterized ncRNAs*

When one or a set of ncRNA sequences belonging to the same class is available, there are various approaches that can be used to identify new or similar members. These methods utilize sequence similarity (homology) or a combination of sequence and structural-based homology searches.

**Sequence-based homology search**

BLAST [4] is a widely used program for finding high-scoring local alignments in a reference database given a single query nucleotide sequence. A few highly conserved ncRNAs in closely related species have been identified using this approach [158, 6, 135]. Compared to protein coding genes, there are many fewer ncRNA sequences available in current databases to run effective BLAST searches. Furthermore, since BLAST comparisons on ncRNAs are performed at the nucleotide level instead of the protein amino acid level, variations in the nucleotide sequence greatly affect the sensitivity of the searches. The short length of ncRNAs also makes it hard to distinguish weakly conserved genes from random hits. Besides, within closely related species, the identification of ncRNAs using this approach has not had much success [31, 146, 81].

If, however, a set of ncRNA sequences belonging to the same class were known, common characteristics such as sequence motifs could be described using profile hidden Markov models (HMM). In profile HMMs, the set of ncRNA sequences are aligned and used to compute a probabilistic model based on nucleotide frequencies from each column. The resulting profile HMM is then used to search a database for new matches similar to the motif in the original set [53]. The reliability of this approach heavily depends on the accuracy of the alignment.

Both BLAST and profile-HMMs have been used to identify protein coding genes; however, the results are not good when applied to ncRNA genes. This poor performance is due to the fast rate of evolution or mutation in ncRNA sequences [114]. Even though the sequences

may change, as long as the change does not affect the structure, the essential function of the ncRNAs is still preserved. For this, another class of methods, which incorporates secondary structure conservation, has been utilized [170].

**Structural-based homology search**

The secondary structure of RNA consists of nucleotide base pairing onto itself. There exists efficient programs based on minimizing the folding energy [175, 65] or maximizing the number of base pairing [24] to predict RNA secondary structure. This secondary structure can be incorporated into the homology search by using a class of models based on context-free grammars (CFG) or the probabilistic version of CFG, stochastic context-free grammar (SCFG), to describe the RNA secondary structure [170]. Given one or a set of aligned ncRNA sequences along with the predicted structure, a model can be generated and used to search a database for new instances [59, 80, 56, 41, 169]. The use of this approach has enabled the computational identification of C/D snoRNA in mammals [168], yeast [99], and other archaea organisms [111]. While useful for predicting homologous ncRNAs, the models are class specific and computationally intensive since they must be optimized for each class [53].

In summary, the above sequence and structural-based homology methods are effective only when the sequence or structure is evolutionarily conserved among different species. The high rate of mutation in ncRNAs compared to protein-coding genes makes it difficult to use sequence homology approaches. Additionally, structural homology approaches fail to detect ncRNAs with little to no RNA structure [40, 13].

#### *1.1.5.2 Methods to predict novel ncRNAs*

A more challenging case is predicting novel ncRNAs where what we are searching for is unknown. Methods that predict novel ncRNAs typically use some combination of comparative analysis, structural information, and genomic signals. We discuss each of these approaches below.

**Comparative analysis based on intergenic regions**

Sequence-based homology methods have also been used to find novel ncRNAs in *E. coli* [173] and other bacterial organisms [118]. This search is done by first extracting sequences between protein coding genes (intergenic regions) and then running BLAST. The protein coding regions are excluded to prevent contamination of false positives from homologous proteins. Intergenic segments conserved among closely related sequences are used with additional information from transcriptional signals [158, 21, 118, 97, 96], structural conservation [8], or gene expression data [158] to identify new ncRNA candidates. The inherent limitations of homology searches have been discussed in a previous section. Additionally, by narrowing the search space to intergenic regions, existing algorithms currently miss all the ncRNAs that may be buried within or overlap protein coding genes. For example, it is well known that about half of C/D snoRNA genes fall within or overlaps protein coding genes [9].

**Structural analysis in conserved genomic alignments**

Similar to structural-based homology methods, approaches in this class work by finding stable secondary structures from sequence alignments. The assumption is that in order to carry out its function, the structure of ncRNA genes must be conserved among two or more organisms. In this case, however, we do not have a set of known ncRNAs to work with but instead rely on genomic regions that are conserved among different species. From these regions, pairwise or multiple sequence alignments can be constructed to identify conserved structures. Various approaches have been used to model the structural information, ranging from analyzing the mutation patterns to examining the RNA structural folding.

One approach to analyzing the mutation patterns of RNA structure uses pairwise alignment between sequences in two related genomes [122, 123, 104]. The alignment is compared to three different probabilistic models for the pattern of mutation. These three models are (*i*) the null model based on a HMM of background base frequencies, (*ii*) the protein coding model characterized by a HMM for codon mutations to preserve the same amino acid code (synonymous codon), and (*iii*) the RNA model as represented by a SCFG, which takes into account the higher rate of substitution in complementary pairs in order to preserve

the secondary structure. Log-odd scores are computed to determine whether the pairwise alignment contains RNA, protein coding genes, or other genomic elements. This approach, as implemented in QRNA, relies heavily on the evolutionary distance between the two sequences used and as such, can suffer from low reliability if the distance lies outside an optimal range [157]. Additionally, the evolutionary distance of the three models used must be the same or else it distinguishes the alignments based on the level of conservation instead of the required pattern of mutation [123].

Other methods search for the conserved RNA secondary structure in multiple sequence alignments [26, 32, 115, 156]. These methods have been primarily applied to vertebrates such as mice, rats, and humans, where large regions of conserved genomic sequences are available. One widely used program is RNAZ [156], which uses conserved multiple alignments without gaps. To identify functionally conserved ncRNAs, a support vector machine is used to combine (*i*) the z-score of the minimum folding energy (MFE) of each individual sequence in the alignment, (*ii*) the structure conservation index defined on the alignment, (*iii*) the mean pairwise identity, and (*iv*) the number of sequences in the alignment. RNAZ and other related programs [26, 32, 115, 156] require the alignments to be precomputed. To compensate for this drawback, other approaches have been proposed to compute both the alignment and secondary structure prediction simultaneously to optimize the structural alignment [147]. Unfortunately, this comes at a large computational cost.

The limitation of this class of approaches is that it requires alignments of at least two or more sequences, which may not always be available. Furthermore, it fails to detect ncRNAs with little RNA structure, such as in the case of some snRNAs and C/D snoRNAs [13].

### Genomic features

All the methods surveyed to this point have relied, to some extent, on comparative genomics. The last class of methods presented in this section is based on using information extracted by the genome itself. Similar to the idea of how genomic signals can be used in protein coding genes, approaches in this branch of novel ncRNA gene finding use information from transcriptional signals, namely, promoters and terminators, along with base composition

variations and other features. A common approach has been to search the intergenic regions to find promoter regions that appear within a short distance of terminator signals [6, 21, 165]. Transcription signals together with specific sequence motif searches in the intergenic regions have discovered specific ncRNAs in more than 60 microbial genomes [83]. The use of these transcriptional elements is limited to only genomes with well-understood regulatory signals. Furthermore, promoter and terminator prediction is neither robust nor comprehensive. For example, current terminator prediction programs can only search for the structurally conserved class of the so-called Rho-independent terminators while missing all the Rho-dependent ones [30]. This limitation imposes constraints on the ncRNA genes and organisms that can be searched.

Base composition methods have had some limited success, primarily in specific organisms where there is bias in the underlying genome. For example, in hyperthermophilic organisms with a genome rich in nucleotide bases `A` and `U`/`T`, the ncRNA genes tend to have high levels of nucleotide bases `G` and `C` [81, 132]. The high (G+C)% is believed to make the ncRNAs more stable in order to withstand high temperatures above 90°C [81]. Besides hyperthermophiles such as *Methanococcus jannaschii* and *Pyrococcus furiosus*, this feature has also been used to identify ncRNAs in the amoeba *Dictyostelium discoideum* [64] and the malaria parasite *Plasmodium falciparum* [146]. The (G+C)% is defined in Eq. (1), where $n_\alpha$ for $\alpha = \{A, C, G, U\}$ is the number of nucleotide $\alpha$ in the sequence.

$$(G + C)\% = 100\frac{n_G + n_C}{n_A + n_C + n_G + n_U} \tag{1}$$

Other compositional measures were also examined [132] but were not found useful. These measures included the compositional difference measures: $(G - C)\% = 100\frac{n_G - n_C}{n_G + n_C}$ and $(A - T)\% = 100\frac{n_A - n_T}{n_A + n_T}$. Another method used to find bacterial ncRNAs [18] has applied mono- and di-nucleotide frequencies together with MFE and known RNA motifs, including $(i)$ structural motifs {`UNCG`, `GNRA`, `CUYG`, `AAR`} and $(ii)$ an observed DNA sequence more common to bacterial ncRNAs {`CUAG`}, where `N` is any nucleotide, `R`=purine={`A`,`G`}, and `Y`=pyrimidine={`C`,`T`/`U`}. The use of mono-, di-, and tri-nucleotide frequencies has also been explored with MFE and a similarity measurement from BLAST to identify ncRNAs in *E.*

*coli* [153]. The main drawback to using compositional-based features is that it may only apply to a small group of organisms, as in the case of (G+C)%.

As mentioned previously, there have been a few attempts to use MFE to distinguish between real and random ncRNAs. Rivas and Eddy [121] embedded a real ncRNA gene into a random sequence with identical mono-nucleotide base composition and found that the z-score of the MFE alone was not statistically significant to be useful in a ncRNA gene finder. Other authors [162] have noted that secondary structure prediction programs compute the MFE by adding stabilizing energy from stacked base pairs with destabilizing energy from loops. Henceforth, shuffled ncRNA must be generated with the same mono- and di-nucleotide composition for any valid conclusions to be drawn about the MFE. Since then, several authors [24, 52] have found that various classes of ncRNAs have MFE significantly lower than its mono- and di-nucleotide shuffled versions. In summary, this result indicates that the MFE of predicted secondary structure is potentially useful for identifying ncRNAs, especially if used together with other features.

Besides the MFE statistics, some authors have proposed some structural folding measures to evaluate the reliability of RNA secondary structure prediction [69, 52]. This helps to identify those ncRNAs that fold more uniquely than their shuffled counterpart. One of these measures is based on the base pairing probability, $P_{i,j}$, for the RNA secondary structure between a nucleotide position $i$ with position $j$ [69]. This probability is assessed using the Shannon base pairing entropy as defined in Eq. (2), where $S_i$ is defined by Eq. (3) and $n$ is the length of the RNA sequence.

$$Shannon\ base\ pairing\ entropy = \frac{1}{n}\sum_{i=1}^{n} S_i \tag{2}$$

$$S_i = -\sum_{j} P_{i,j} log(P_{i,j}) \tag{3}$$

Although never applied for genome-wide prediction, the Shannon base pairing entropy and other folding measures have been evaluated to test their ability to discriminate between various classes of ncRNA and its shuffled sequences [52]. Another recent program called coding or non-coding [94], CONC, uses known protein features to help distinguish between proteins and ncRNAs. The set of features used includes peptide (amino acid) length, amino

acid composition, percentage of residues in certain predicted protein secondary structures, percentage of exposed residues, compositional entropy, number of homologs in database search, alignment entropy, nucleotide frequencies (mono, di, tri), and average hydrophobicity of amino acid residues. This method was applied to detect ncRNAs in certain eukaryotic organisms. The recent growth of genomic-based features for ncRNA gene finding is promising; however, more work remains to be done.

### 1.1.6 Operon prediction

Furthermore, we choose to examine higher-order genome organization because little is known about the transcriptional mechanism of ncRNA genes. By examining the position of ncRNAs in relation to these genomic structures, we are able to infer information about how it is transcribed (i.e., whether it is transcribed together with operons or not). The idea is that if it is transcribed together with operons, we can use knowledge of operon structures to help narrow the search space. Genome annotation of operons enables a more thorough understanding of the higher-level organization of genes.

The operon is the most basic level of gene organization in prokaryotes and consists of two or more protein-coding genes that are transcribed together on a single mRNA transcript. The characterization of operons represents an important step in understanding many cellular processes and deciphering transcriptional regulatory networks [145]. Due to the arduous nature of experimentally determining operons on an individual basis, there is a need for computational approaches for operon prediction.

**Table 1.2:** Overview of common features used in operon prediction methods.

| | | |
|---|---|---|
| Comparative analysis | Phylogenetic profiles | [23] |
| | Conserved gene order across different genomes | [46, 22, 120, 161, 154] |
| Sequence features | Intergenic distances | [131, 107, 27, 23, 161] |
| | Codon usage | [12, 120] |
| | Short DNA motifs | [28] |
| | Transcriptional signals (promoter, terminator) | [166, 27] |
| Functional annotation | Pathways | [174, 126, 71] |
| | Expertly curated information | [131] |
| | Gene Ontology (GO) similarity | [28] |
| | Clusters of Orthologous Genes (COG) class | [23, 154] |
| Experimental data | DNA microarray | [27, 128, 12, 29] |

Existing operon prediction methods use information from comparative analysis, sequence features, functional annotation, and experimental data, as summarized in Table 1.2. Of particular interest are three operon predictors considered to be better methods among all publicly available operon prediction programs. The first program is Joint Prediction of Operons (JPOP) [23], which trains a neural network on three features: intergenic distance, similarity between their phylogenetic profiles, and relatedness of their annotated functions from clusters of orthologous genes (COG) [143]. The second program, Operon Finding Software (OFS) [161], applies a naïve Bayes method on conserved gene-order information across multiple genomes together with intergenic distance and similarity information of annotated gene functions to make operon predictions. The third program, developed by the Virtual Institute for Microbial Stress and Survival (VIMSS) [120], is similar to the first two programs because it uses intergenic distance and COG information. VIMSS differs, however, in also employing the codon adaptation index (CAI) and applying a different approach for comparative genome analysis to make operon predictions. The results are incorporated into a naïve Bayes approach to make predictions.

### 1.1.7 Summary

Within the past decade, the main approaches used for ncRNA gene prediction have been based on comparative genomics. The inherent limitation of these approaches lies in their difficulty in finding ncRNA genes not conserved among closely related species. Techniques to identify novel ncRNAs using comparative analysis of intergenic regions limit the available search space and thus prevent the identification of ncRNAs contained in or overlapping protein coding regions. Methods relying on multiple alignments to search for stable ncRNAs add another level of complexity in requiring the existence of conserved alignments. Approaches based on transcriptional and compositional features have had some limited success, although their application to all organisms is questionable. In general, there is a need for non-homology-based methods to expand upon the number of ncRNAs able to be detected. The exploration of more features to better characterize ncRNAs and the understanding of the role of higher genomic structures are needed in order to apply the

search universally across a large set of organisms. This investigation is at the focus of our dissertation work.

### 1.1.8 Organization of this Dissertation

The dissertation is organized as follows:

In Chapter 2, we will present a novel neural network-based meta-learner for operon prediction [145]. This method has been applied to predict operons in the bacteria *Escherichia coli* and *Bacillus subtilis* and the hyperthermophilic archaeon *Pyrococcus furiosus*. Knowledge of the operon organization will enable us to study the context of ncRNA genes and better understand its transcriptional environment.

Next, in Chapter 3, we will present a *de novo* computational method for predicting ncRNA genes. We have developed a data set of ncRNA genes obtained from literature and existing databases. We also identify unique features inherent to ncRNAs in order to develop a machine learning classifier to predict ncRNAs on a genome-wide scale.

Then, in Chapter 4, we will discuss some additional applications of our computational method for predicting ncRNA genes. We present the relationship found between ncRNAs and operons.

Finally, in Chapter 5, we will summarize this dissertation and suggest topics for future research. To aid readability, we have attempted to keep every chapter as self contained as possible.

# CHAPTER II

# OPERON PREDICTION IN *PYROCOCCUS FURIOSUS*[1]

To provide insight into the transcriptional mechanisms of non-coding RNA genes, we examine higher order genomic structures, namely operons. The ability to predict operons enhances our knowledge of gene regulation and function. Our later studies of non-coding RNA genes in the context of operons will provide additional insight on how these non-coding RNA genes are transcribed to determine whether there is a preference for these genes to be independently transcribed or whether they are transcribed with operons. In this chapter, we explore a novel meta-learner approach for operon prediction.

## *2.1 Abstract*

Identification of operons in the hyperthermophilic archaeon *Pyrococcus furiosus* represents an important step to understanding the regulatory mechanisms that enable the organism to adapt and thrive in extreme environments. We have predicted operons in *P. furiosus* by combining the results from three existing algorithms using a neural network. These algorithms use intergenic distances, phylogenetic profiles, functional categories, and gene order conservation in their operon prediction. Our method takes as inputs the confidence scores of the three programs, and outputs a prediction of whether adjacent genes on the same strand belong to the same operon. In addition, we have applied Gene Ontology (GO) and KEGG pathway information to improve the accuracy of our algorithm. The parameters of this neural network predictor are trained on a subset of all experimentally-verified operon gene pairs of *B. subtilis*. It subsequently achieved 86.5% prediction accuracy when applied to a subset of gene pairs for *E. coli*, which is substantially better than any of the three prediction programs. Using this new algorithm, we predicted 470 operons in the *P. furiosus* genome. Of these, 349 were validated using DNA microarray data.

[1]This chapter was published in [145] and is a result of joint work with Phuongan Dam, Zhengchang Su, Farris L. Poole, II, Michael W. W. Adams, G. Tong Zhou, and Ying Xu.

## *2.2 Introduction*

*Pyrococcus furiosus* is a hyperthermophilic anaerobic archaeon that grows optimally near 100°C using carbohydrates and peptides as carbon and energy sources [48]. This organism is commonly found in hydrothermal vents on the seafloor near volcanos. Its ability to grow to high cell densities under laboratory conditions without the need of elemental sulfur, and thus production of toxic hydrogen sulfide, has made it a useful model organism with which to study thermostable enzymes and adaptations to high temperature environments [1]. The genome sequence of *P. furiosus* has been determined [124, 119] and is available at http://www.ncbi.nlm.nih.gov/entrez/viewer.fcgi?db=nucleotide&val=18976372. The latest annotation contains 2,125 genes, which is used for the study herein. Like most genome annotations, that of *P. furiosus* provides a unique source of information for molecular and biochemical studies, but it is mostly gene-centric and does not provide much structural and functional information about higher-level organizations. In this chapter, we present our recent work on the identification of operons in *P. furiosus*.

An operon is defined as a basic transcriptional unit in prokaryotes. Characterization of operons represents an important step in understanding many cellular processes and deciphering transcriptional regulatory networks. Insights into the function and regulation of genes in the context of pathways and networks can be gained if we can annotate operons accurately. Due to the arduous nature of experimentally determining operons on an individual basis, several computational approaches have been proposed for predicting them [23, 120, 46, 161, 174, 22]. Generally, these approaches use information derived from comparative genomics, transcriptional signals up- and downstream of operons, features such as intergenic distances, functional annotation of genes, and experimentally-derived DNA microarray data.

We have recently developed a novel method for operon prediction by integrating three existing operon-prediction methods and have applied it to the bacteria *Escherichia coli* and *Bacillus subtilis*, and *P. furiosus*. The three methods were chosen because they are considered as better prediction methods among all publicly available operon prediction programs.

All three prediction programs assume that genes within an operon are in the same directon, which is defined as consecutive open reading frames (ORFs) transcribed in the same direction with no intervening ORF on the opposite strand [23, 120, 161]. The first program is the JPOP (Joint Prediction of Operons) program [23], which classifies each pair of consecutive genes as an "operonic" or an "non-operonic" boundary based on their intergenic distance, similarity between their phylogenetic profiles, and relatedness of their annotated functions from COGs [143]. Each of these sets of supporting data is integrated using a neural network to generate operon predictions. The second program, Operon Finding Software (OFS) [161], combines conserved gene-order information across multiple genomes with intergenic distance and similarity information of annotated gene functions to make operon predictions. This work generalized the gene order conservation approach used in [46] by relaxing the adjacency and orthology criteria. The authors of OFS [161] claimed to be able to predict operons without extensive training. The third program, developed by the Virtual Institute for Microbial Stress and Survival (VIMSS) [120], is similar to the first two programs because it uses intergenic distance and COG information. VIMSS differs, however, in also employing the codon adaptation index (CAI) and applying a different approach for comparative genome analysis to make operon predictions. The comparative genome analysis examines how often orthologous genes are close to each other within 5 kb across multiple genomes, while the CAI measures synonymous codon usage. Another operon prediction method developed by The Institute for Genomic Research (TIGR) [46] was considered but not integrated into our method due to the high number of missing confidence values between adjacent gene pairs in the same directon. A summary of the default operon prediction results for the three programs for *E. coli* and *B. subtilis* are shown in Figure 2.1. The Venn diagram displays the number of gene pairs predicted to be within operons by each program and the overlap in gene pairs predicted among the three programs. Out of a total of 2,985 adjacent gene pairs in the same directon in *E. coli*, 1,885 gene pairs are predicted to be in operons by at least one program and only 55% (=1037/1885) of gene pairs are predicted to be in operons by all three programs. Likewise in *B. subtilis* with a total of 3,005 gene pairs, 2,122 gene pairs are predicted to be in operons, but only 28% (=599/2122) of gene

pairs are predicted by all three programs. With low consensus among individual operon prediction programs, there is a need to incorporate the additional information provided by each program into a general operon predictor.

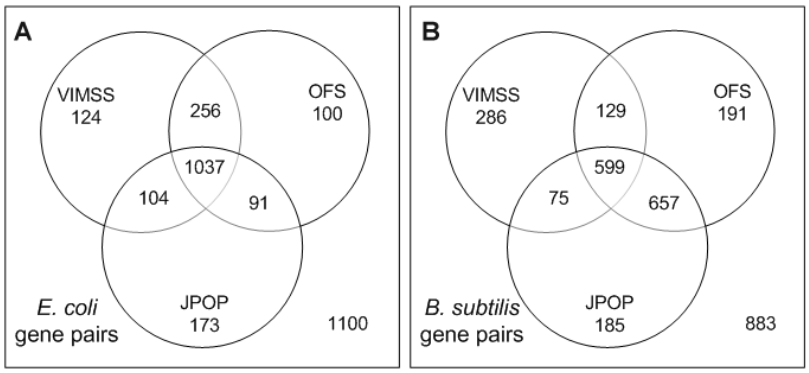


**Figure 2.1:** Venn diagrams of adjacent gene pairs in the same directon predicted to be operon gene pairs by the three prediction programs: JPOP, OFS, and VIMSS for (A) *E. coli* and (B) *B. subtilis*. The operon predictions are the result of using an optimal threshold that maximizes the (Sensitivity+Specificity) value fixed from *B. subtilis* training for all programs. In (A), 1,885 (= 124 + 256 + 100 + 1037 + 104 + 91 + 173) gene pairs are predicted to be operon gene pairs by at least one prediction program. Likewise in (B), this number is 2,122 (= 286 + 129 + 191 + 599 + 75 + 657 + 185) gene pairs. Examining those operon gene pairs in common from all three prediction programs, there is only a 55% (= 1037/1885) and 28% (= 599/2122) overlap in predicted gene pairs in *E. coli* and *B. subtilis*, respectively, indicating little consensus among the three programs.

Our initial prediction stems from training a neural network-based classifier (to classify a pair of adjacent genes as either operonic boundary or not), based on the outputs of the three aforementioned programs. Furthermore, we use additional computational data from (a) Gene Ontology (GO) information, (b) known pathway information, and (c) log-likelihood intergenic distance to improve the operon prediction accuracy. The GO classification is used to compute a functional similarity score between pairs of adjacent genes in the same directon. Additionally, we have computed KEGG pathway scores based on whether or not gene pairs belong to common KEGG pathways. The intergenic distance feature as used in previous studies is also inputted directly into the neural network to aid prediction since it has been found to be a strong discriminatory feature [23, 107, 131]. Three-fold cross-validation and train/test set validation was analyzed for *E. coli* and *B. subtilis* to examine the performance of these features within and across species, respectively. Using the optimal

training set, our method is applied to make operon predictions in *P. furiosus*. We have used experimental data obtained from time-course microarray gene expression data to verify these operon predictions. The idea is that genes in the same operon should in general exhibit similar expression patterns under any experimental conditions. All predicted operons are available at http://csbl.bmb.uga.edu/∼tran/operons along with the prediction program.

## *2.3 Materials and Methods*

We present our method by first introducing how the positive and negative sets for training and testing were generated. Then, we discuss the various features used to train the operon predictor: confidence scores from JPOP, OFS, and VIMSS, GO similarity scores, KEGG pathway scores, and intergenic distances. Next, we explore the design of our neural network (NN) based predictor. Finally, we examine how we use the available microarray data to validate our predictions in *P. furiosus*.

### 2.3.1 Generation of the positive and negative data sets

Since no genome-wide operons in *P. furiosus* have been experimentally determined, we have benchmarked our program using operons from *E. coli* and *B. subtilis*, which have been experimentally validated. The true positive (TP) set in *E. coli* are the transcriptional unit gene pairs extracted from the RegulonDB database [130]. The generation of the negative set represents a challenge in this study since current operon prediction programs typically only output the confidence score of adjacent gene pairs in the same directon. Defining a negative set using gene pairs from opposite strands as used in [23] is not applicable to our approach because the confidence scores are not defined. These gene pairs are considered trivial by current prediction methods since opposite strand information alone is enough to classify them. Furthermore, using an intergenic-distance cutoff to generate a negative set may impose certain biases and prevent the identification of operonic gene pairs that are located far away. We generate our negative data set as follows: two adjacent genes in the same directon are considered as a true negative (TN) gene pair if they are not transcriptionally co-expressed, i.e., not in the same transcriptional unit. These TN gene pairs also include adjacent genes not in a transcriptional unit with one that is present in a transcription unit.

This approach has been used by other authors [107, 73, 71, 46, 44]. Operons of *B. subtilis* obtained from [30] were extracted similarly. In addition, we consider only gene pairs with confidence measures from all three prediction programs when generating the true positive and true negative sets. The number of TP gene pairs in *E. coli* and *B. subtilis* are 711 and 628, respectively. The number of TN gene pairs in *E. coli* and *B. subtilis* are 374 and 556, respectively.

In addition to the above two sets, we also consider the whole operonic gene pair set (TP set plus those without confidence scores defined by all three prediction programs) and a non-operon set defined as pairs of adjacent genes, with one gene on one strand and the other gene on the opposite strand. We refer to these sets as operon and non-operon gene pair sets, not to be confused with the true positive (TP) and true negative (TN) sets as defined earlier. The operon set contains 821 gene pairs in *E. coli* and 806 gene pairs in *B. subtilis*, while the non-operon set contains 1,256 gene pairs in *E. coli* and 1,099 gene pairs in *B. subtilis*. By having the non-operon set, we know for sure that gene pairs from the opposite strands do not belong to the same operon. This allows us to detect any biases induced by our definition of the TN set when we examine the different features.

### 2.3.2 Operon prediction by existing software

Executable codes for JPOP were obtained from http://csbl.bmb.uga.edu/downloads/ [23]. The software for OFS was downloaded from http://www.cse.wustl.edu/∼jbuhler/research/operons. We have applied OFS using the default values for all parameters and $\beta = 0.35$ as used in [161]. VIMSS had precompiled operon predictions for some organisms, available at http://www.microbesonline.org/operons [120]. Perl scripts were written to extract the confidence values for each gene pair in the TP and TN set. For all organisms studied in this paper, the focus is on operons with two or more genes.

### 2.3.3 Gene Ontology (GO) similarity analysis

One additional source of evidence used to improve our neural network predictor is the GO classification [7], which encompasses three levels of biological functions: biological process,

molecular function, and cellular component. It is known that genes in the same operon are involved in the same or similar biological processes; hence GO ontology information should in principle be helpful for operon prediction. A similarity score is used as defined in [163], which examines the GO-based functional assignments for each gene pair and uses an acyclic graph [160] in the form of a tree structure to compute the depth of the common terms. The higher the score, the more similar the two genes are since they share more common GO terms.

The distribution of GO similarity scores for operon and non-operon gene pairs of *E. coli* and *B. subtilis* are shown in Figure 2.2. We also include the distribution for the {TP,TN} set in order to compare with the operon,non-operon set. As shown in Figure 2.2, the non-operon and TN gene pairs tend to have lower GO similarity scores compared to operon and TP gene pairs. This capability to discriminate between operon and non-operon gene pairs makes this feature useful in our neural network predictor. The coverage of GO similarity scores with annotation in *E. coli*, *B. subtilis*, and *P. furiosus* is 50%, 45%, and 29%, respectively.

### 2.3.4 Pathway assignment

Genes within the same operon generally encode proteins that function in the same metabolic pathway or biological process. As such, we have generated scores based on the pathway information collected from the Kyoto Encyclopedia of Genes and Genomes (KEGG) database [77]. The KEGG (prokaryotic) pathways are organized into four general categories (level 1): Metabolism, Genetic Information Processing, Environmental Information Processing, and Cellular Processes. These categories are further subdivided into level 2 and level 3 pathways. The authors of KEGG have used KEGG Orthology (KO) terms to describe each pathway. Given a genome with annotated genes, one can assign KO terms to each gene and determine directly the pathway in which it is involved. The KO terms can be obtained for each organism using the following methods:

1. KEGG Orthology (KO) flatfile contains manually curated data linking KO labels to genes of various organisms. This flatfile was obtained from

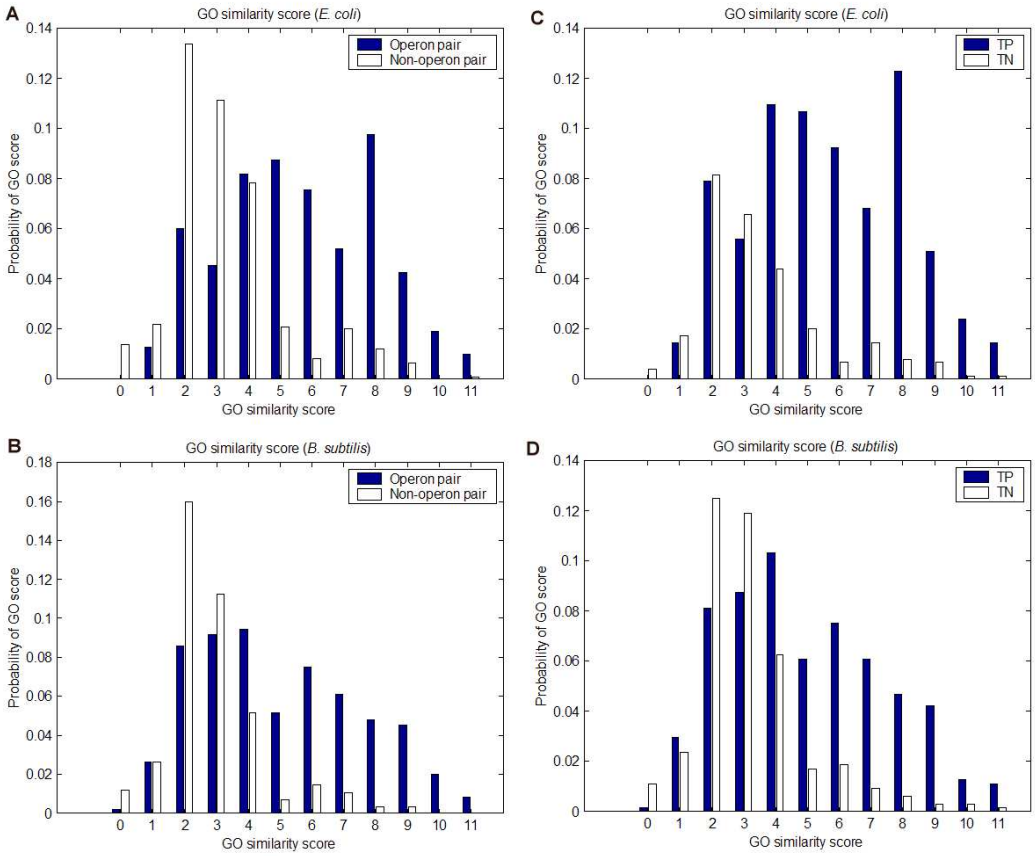


**Figure 2.2:** Distribution of GO similarity scores. The probability distribution for operon and non-operon pairs is plotted over the range of possible GO similarity scores for (A) *E. coli* and (B) *B. subtilis*. Non-operon gene pairs tend to have lower similarity scores compared to operon gene pairs. The probability distribution for the true positive (TP) and true negative (TN) gene pairs used in the validation studies are given for (C) *E. coli* and (D) *B. subtilis*. A similar trend is observed in which the majority of TN gene pairs clusters around lower GO similarity scores compared to TP gene pairs.

ftp://ftp.genome.jp/pub/kegg/tarfiles/ko and processed to obtain all genes in the organism with KO annotation.

2. KO-Based Annotation System (KOBAS) software [100] was used to annotate the KO terms based on sequence-similarity searches. The program uses BLAST to assign KO labels to genes with an E-value $< 10^{-5}$ along with some additional information.

3. KEGG Automatic Annotation Server (KAAS) was also used to annotate the genes. This annotation uses a bi-directional best hit of BLAST with a threshold BLAST bit score $> 60$.

The results of these three programs were combined using the following simple heuristic rule: when the annotation results are different among the three different methods, we use the result by the method with the highest priority score. The KO flatfile annotation is given the highest priority score since the results are manually curated. KOBAS is given the second highest priority since its annotation agrees with the KO flatfile more often than KAAS. Lastly, if the KO flatfile and KOBAS cannot assign a gene to a pathway, the annotation result from KAAS is used. By using multiple sources to assign the KO terms, we have increased the coverage of ORFs with KO annotation. In *E. coli*, the numbers of ORFs in the KO flatfile represents 55% of all protein-coding ORFs. By adding similarity search annotation methods like KOBAS and KAAS, the coverage improves to 61%. Likewise for *B. subtilis*, the coverage improves from 46% to 54%. For *P. furiosus*, where the number of manually curated ORFs in the KO flatfile represents only 38% of all protein-coding ORFs, the addition of similarity search methods improves the coverage up to 46%.

Once the ORFs are assigned KO annotation, the KEGG pathways can be inferred directly. A KEGG pathway score of 1, 2 or 3, was assigned to a gene pair if they share the same level 1, level 2, or level 3 pathway, respectively. The higher this score, the higher the chance the two gene products are in the same pathway, and hence it is more likely that the two genes belong to the same operon. A score of -1 was assigned to a gene pair if none of them have pathway annotation while a score of 0 was assigned if only one gene has pathway annotation. The score distribution for the operon, non-operon and TP, TN data sets for

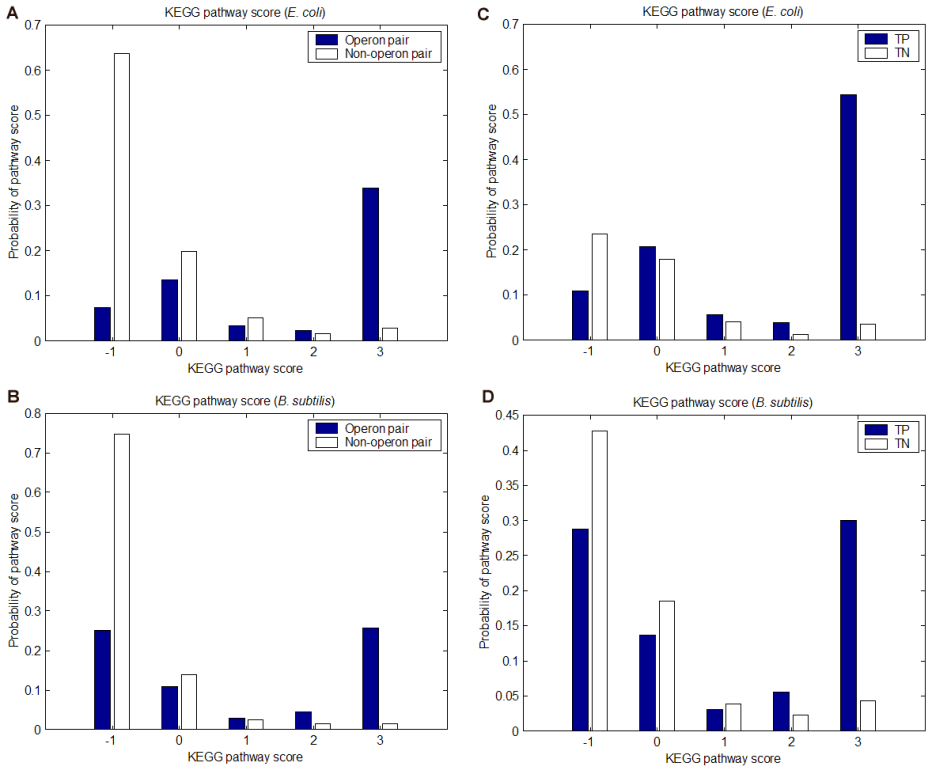


**Figure 2.3:** Distribution of KEGG pathway scores. The probability distribution for operon and non-operon pairs is plotted over the range of possible KEGG pathway scores for (A) *E. coli* and (B) *B. subtilis*. A score of 1, 2, or 3 indicates that a gene pair shares common level 1, level 2, or level 3 KEGG pathway, respectively. A score of 0 indicates that KEGG pathway annotation is only available for one of the gene pairs. A score of -1 indicates that KEGG pathway annotation is not available for both of the gene pairs. Operon gene pairs typically have a KEGG pathway score of 3 (i.e., share the same level 3 KEGG pathway). The probability distribution for the true positive (TP) and true negative (TN) gene pairs used in the validation studies are given for (C) *E. coli* and (D) *B. subtilis*.

*E. coli* and *B. subtilis* are shown in Figure 2.3. Operon gene pairs typically have the same level 3 pathway.

### 2.3.5 Intergenic distance-based log likelihood score

Another input to our NN-based predictor is an intergenic distance-based log likelihood score defined by Eq. 4. This score for a gene pair is computed as the log ratio between the probability that their distance belongs to the distance distribution of the TP set and the probability this distance belongs to the distance distribution of the TN set [131]. Although this feature is already included in JPOP, OFS, and VIMSS, adding it to the input of the

neural network predictor is shown to improve the prediction in some cases. For our current study, the intergenic distances were computed based on the training sets and applied to score the distances on the test set. This is necessary since the true positive and true negative distributions are not known. The histogram of intergenic distances for the different sets is shown in Figure 2.4. There is a clear peak separation between the {TP,TN} set in *B. subtilis* whereas in *E. coli*, the TN set is less pronounced and has approximately a uniform distribution. This suggests that the performance in using the intergenic feature will vary between *B. subtilis* and *E. coli* based on our choice of the TP and TN set. In the general case; however, the intergenic feature is beneficial in discriminating between the operon and non-operon gene pairs.

$$LL(d(g_a, g_b)) = ln \frac{P(d(g_a, g_b)|TP_{genepair})}{P(d(g_a, g_b)|TN_{genepair})} \tag{4}$$

### 2.3.6 Neural network-based operon predictor

An artificial neural network was implemented to integrate the outputs from three operon prediction programs in order to attain a more robust and efficient tool for operon prediction. Intuitively, by utilizing and consolidating the strengths of these programs into a single operon prediction tool, improved results can be realized. The question is how to best combine the programs to produce the best possible prediction results. Neural network is a proven technique for combining multiple sources of information, without assuming the underlying relationships among the individual data sources. This technique is robust for noisy data and has been widely used for many biological data analysis problems [18, 167]. The design of our NN-based predictor is achieved through three main steps: (a) data pre-processing (feature extraction and normalization), (b) selection of appropriate network architectures (e.g., number of layers, number of neurons), and (c) training and testing. Iterations of (b) and (c) are needed to identify the best network architecture based on the prediction performance.

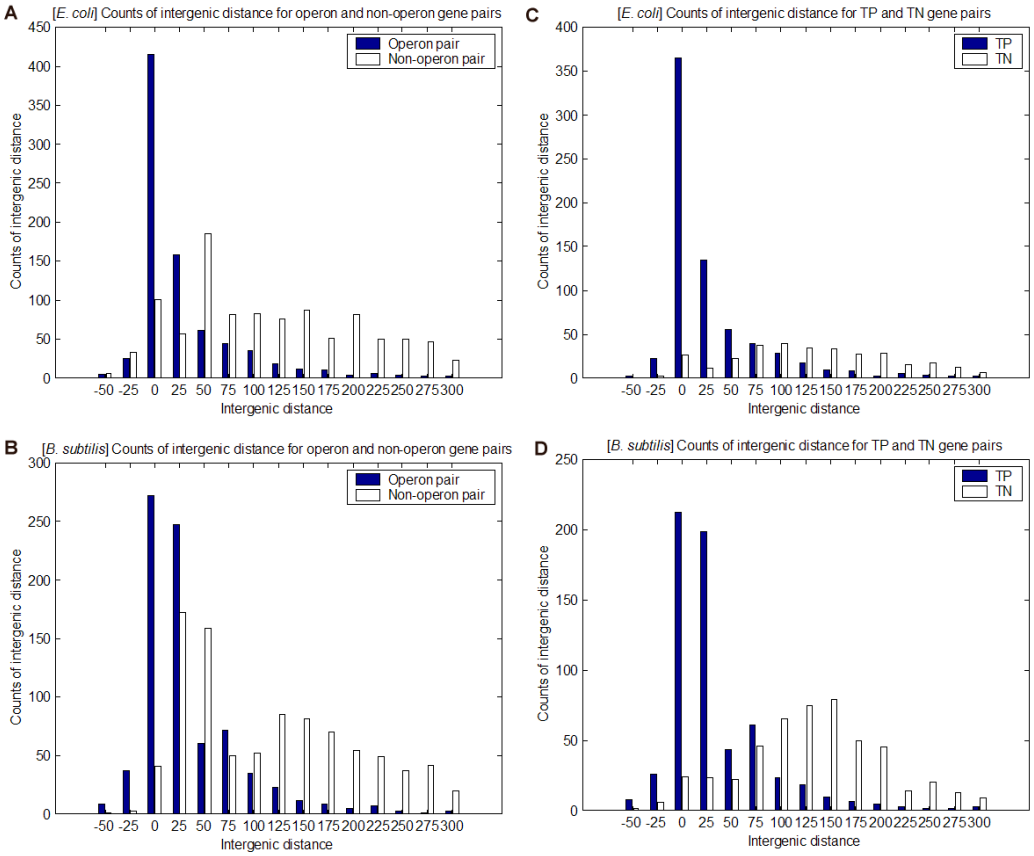


**Figure 2.4:** Histogram of intergenic distances. The counts of intergenic distance for operon and non-operon pairs are plotted over the intergenic distance range of -50 to 300 nt for (A) *E. coli* and (B) *B. subtilis*. The same is done for that of the true positive (TP) and true negative (TN) set for (C) *E. coli* and (D) *B. subtilis*. Comparing (C) and (D), the distribution of the intergenic distance for *B. subtilis* has two well-defined peaks whereas in *E. coli*, the TN distribution is more uniform. This is due to the inherent property of the TN set in these two organisms. As discussed in the results, using the log-likelihood of the intergenic distance in the *B. subtilis* data set improves performance more than in the *E. coli* data set because its distribution is more "discriminative".

#### *2.3.6.1 Score normalization*

Normalization of the confidence scores of the three prediction programs is needed to ensure that the dynamic range of the individual programs does not influence the performance of the neural network. For each program, the prediction confidence measure for each gene pair was extracted and normalized to between 0 and 1, where a value above 0.5 indicates that the corresponding gene pair belongs to the same operon. A linear mapping was performed to recenter the JPOP confidence scores between [0 1]. This scaling was used to keep it consistent with the probability-based confidence measures from VIMSS and OFS. The GO similarity score, KEGG pathway scores, and log likelihood score of intergenic distance were also linearly normalized into the range [0 1].

#### *2.3.6.2 Neural network training*

The idea behind a neural network is to train a set of parameters to give a desired output target ($t$), for a given input data ($x$). A trained network can then be applied to new data $x'$ to predict the outcome $t'$. In our approach, we present the confidence measures from each of the three prediction programs, $x = [x_i]$ for $i = 1, 2, 3$ to a feed-forward network architecture (see Figure 2.5). Various combinations of GO similarity, KEGG pathway, and intergenic distance scores were also tested as additional inputs into the NN-based predictor. The desired output target is 0/1 {1= "gene pair in an operon", 0= "gene pair not in an operon"}. The training algorithm optimizes the weights $W = [w_i]^T$ and a bias, $b$, of the network during the training (supervised learning) phase to minimize the error between the network output, $a$, and the desired output, $t$, on the training data. Our neural network was trained using MATLAB®'s neural network toolbox. The network parameters are optimized using the Levenberg-Marquardt algorithm. Other network training functions were tested but either the results are not as good or the differences are insignificant compared with our selected network (results not shown). The network architecture parameters are (a) the transfer function ($f$), (b) the number of neurons, and (c) the number of layers. Various network architectures were tried and tested on the *E. coli* and *B. subtilis* data sets. Based on the performance on *B. subtilis* data, an optimal network architecture is selected and

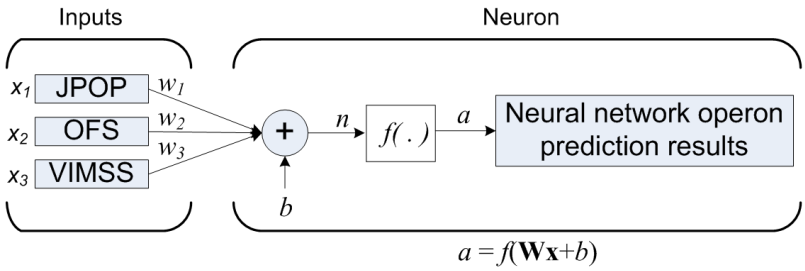


**Figure 2.5:** Schematic illustration of a one-layer neural network architecture with three inputs from existing programs. The confidence values $x_i$ of each operon prediction program are inputs into a neuron consisting of a summation unit and a transfer function, $f$, to produce an output $a$.

applied to predict operons in *E. coli* and *P. furiosus*.

Plots of the receiver operating curve (ROC) [47] were generated to compare our operon predictor with the three programs. The ROC plots the sensitivity versus (1-specificity) over a range of program thresholds. For each classifier, the area under its receiver operating curve (AUROC) can be computed to give a qualitative measure of the performance not dependent on a specific threshold. Generally, the higher the AUROC, the closer the classifier's ROC is to the "optimal" performance, i.e., higher sensitivity and higher specificity. We have also calculated the sensitivity, specificity, and accuracy values of our predictions, as defined by Eqs. 5, 6, and 7, where TP is the true positive, TN is the true negative, FP is the false positive, and FN is the false negative. The overall prediction accuracy takes into consideration both the number of true positive and true negative correctly predicted to provide for a good comparison of the performance among the individual programs.

$$Sensitivity = Sn = \frac{TP}{TP + FN} \tag{5}$$

$$Specificity = Sp = \frac{TN}{FP + TN} \tag{6}$$

$$Accuracy = \frac{TP + TN}{TP + FN + FP + TN} \tag{7}$$

### 2.3.7 Microarray data analysis

Microarray gene expression data can be used to verify the operon prediction in *P. furiosus*. Genome-wide cDNA microarray data representing the original 2,065 genes in the *P. furiosus* genome have been published in response to changes in carbon source [136] and cold shock [159]. Available microarray data in *P. furiosus* are analyzed by two methods. The first one applies a correlation approach to cluster time series microarray data. It is expected that genes within an operon exhibit similar trends in their expression profiles. The second method analyzes all published microarray data sets to identify significantly expressed gene pairs. This approach has been applied in [136] and [159] to identify groups of putative operons. We have re-analyzed the raw data here in order to standardize the preprocessing procedure among the data sets.

#### *2.3.7.1 Kinetic cold shock-microarray data and application*

In this method, the expression trends of genes within an operon over time are analyzed by using available time series (kinetic) cold shock microarray data. In the kinetic cold shock experiment [159], the organism was grown at 95°C and then subjected to cold shock at 72°C starting at time $t = 0$. The intensity values of the cDNA microarray were monitored at time $t = 1, 2$, and 5 hour(s). The kinetic cold shock experiment consisted of two replicates done on different dates. Each replicate consists of two duplicate slides with each having three copies of the ORF spotted on the cDNA array. The copy, duplicate, and replicate terminology is illustrated in Figure 2.6. The raw data were preprocessed as follows. Any signal or reference data point with an intensity value <2000 was set to 2000 as values under this cutoff are considered too low to be significant. Reference intensities are the initial condition immediately before cold shock at $t = 0$. The $log_2$(signal/reference) ratio is then computed where a positive value indicates up regulation, zero indicates no regulation, and a negative value indicates down regulation. An averaging step then averages the $log_2$ ratios of duplicates with copies that are all nonzero. Non-zero duplicates within the same replicate were averaged and then combined with non-zero replicates to generate a single average $log_2$ ratio to represent the expression of each ORF for time $t = 1, 2$, and 5 hour(s), respectively.

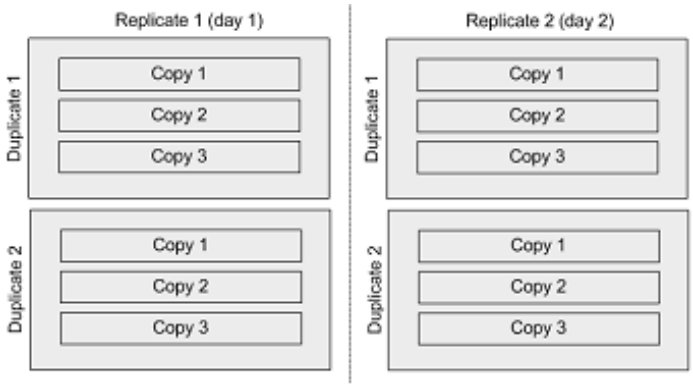


**Figure 2.6:** Experimental setup of the kinetic cold shock microarrays for *P. furiosus*. The kinetic cold shock experiment consists of two replicates done on separate dates. For each replicate, there are two duplicates on two separate slides. For each duplicate, there are 3 copies of each ORF where it is spotted on the cDNA array. Having these multiple copies and duplicates helps average out errors due to slide contamination and fabrication while having multiple replicates helps average out experimental variability.

The Pearson correlation coefficient $r$ as defined in Eq. 5, where $x = [x_{t0}, x_{t1}, x_{t2}]$ and $y = [y_{t0}, y_{t1}, y_{t2}]$ represent the time series profiles for adjacent ORF $x$ and ORF $y$ at $t_0$ = 1 hour, $t_1$ = 2 hour, and $t_2$ = 5 hour. Adjacent ORFs of the same directon with a Pearson correlation coefficient >0.5 and intergenic distance $\leq$ 75 bp are predicted to be in the same operon. The intergenic distance cutoff was determined from examining a list of 33 known/putative operons published in literature [151, 136, 159, 82] where the longest intergenic distance between an operonic gene pair was 74 bp. Using the correlation coefficient and intergenic distance cutoff, we generate a "microarray evidence list" consisting of 357 operons to validate the operon predictions of *P. furiosus*.

$$r(x,y) = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2 \sum_{i=1}^{n}(y_i - \bar{y})^2}} \tag{8}$$

*2.3.7.2 Generation of putative operon list based on microarray data*

To date, the only studied operons in *P. furiosus* have been the lamA [151] and POR/VOR [82] operons, both involved in energy metabolism. Several putative operons have been suggested in [136] and [159] based on peptide+sulfur versus maltose+sulfur and batch/kinetic cold shock data, respectively. The raw data values for these microarray experiments were obtained from the original authors and the preprocessing was standardized among all of

these data sets. A one-sample, two-sided unpaired t-test was performed on the $log_2$ ratios to identify differentially expressed ORFs and the p-value was determined by applying the Holm's step-down correction similar to the original papers. Unlike the original paper, the average fold change was computed using only non-zero $log_2$ ratios. This results in less bias of the fold change towards lower values due to the inclusion of noisy data points. Adjacent ORFs with an average fold change $\geq 2$, a Holm's adjusted p-value $\leq 0.01$, and an intergenic distance $\leq 30$ bp were predicted as putative operon gene pairs. We refer to this list of operons as the "putative operon list". An intergenic distance cutoff of 30 bp is used, as it is small enough so as not to contain any internal promoters such as TATA-boxes known to be upstream of the transcriptional start site.

## *2.4 Results and Discussion*

Our approach seeks to produce operon predictions with a higher accuracy (overall correct predictions) compared to any of the three individual programs. In order to find the optimal set of input features, we first run 3-fold cross-validation on *E. coli* and *B. subtilis*. Using these features, we evaluate the performance across species by training on one organism and applying it to another. After predicting the operons for *P. furiosus*, we apply the microarray results to identify a putative list of operons to do further experimental study. We describe the results of each procedure in the following sections.

### 2.4.1 Cross-validation on *E. coli* and *B. subtilis*

A 3-fold cross-validation was performed on *E. coli* and *B. subtilis* data using various combinations of inputs into the neural network. From the ROC curves, the AUROC was computed to compare the performance of our NN-based predictor with the three existing programs as summarized in Table 2.1. For comparison and reproducibility, we fix the network to be a single neuron 1-layer network with a transfer function $f$=logsig. This network will be used unless otherwise specified. By integrating the confidence scores of the three existing programs, we are able to achieve prediction results better than any single program. Even better performance was achieved by incorporating GO, pathway, and the intergenic distance information. For *E. coli*, the most useful features (in order of most to least helpful) are

pathway, GO, and intergenic distance. On the other hand, for *B. subtilis*, the intergenic distance was most beneficial, while the pathway and GO information achieved comparable performance. As expected, using the two most helpful features for both organisms produced slightly better results. The two sets of features that performed best can be observed in Figures 2.7 and 2.8. At virtually all threshold levels, the NN-based predictor achieves both higher sensitivity and higher specificity than any of the three programs. For *E. coli*, adding the intergenic distance actually decreased the performance compared to using inputs from the three programs with GO and pathway information. In this case, the intergenic distance adds more noise rather than helping. This can be explained by examining the histogram of the intergenic distance for *E. coli* compared with *B. subtilis* on the {TP,TN} set as shown in Figure 2.4. While the best performance for *E. coli* was achieved using additional GO and pathway information, *B. subtilis* benefited from using all three additional features. This is analyzed in further detail in the next section. As a side note, increasing to a two-layer neural network with more neurons only improves the AUROC by less than 0.01 (results not shown).

### 2.4.2 Validation on *E. coli* using *B. subtilis* training set

Based on the optimal set of input features determined in the previous section, we train our NN-based predictor using the entire *B. subtilis* data set and test on *E. coli*. The results of comparing the performance of the NN-based predictor (3 programs + GO + pathway) with and without the intergenic distance are shown in Table 2.2. The overall accuracy on the test set of the NN-based method is higher than any of the three existing programs. Since the test accuracy of the NN-based method is comparable with or without the use of intergenic distance, we decide to use the NN-based predictor with the six inputs including the intergenic distance because of the observed higher training accuracy. Using this architecture, the neural network was further optimized to a 2-layer (2 hidden neurons, 1 output neuron) to improve the overall accuracy of the test set from 0.8544 to 0.8645 as shown in Table 2.2. The sensitivity of our NN-based predictor is comparable to OFS and JPOP; however, there is over 6% to 8% improvement in the specificity, respectively. For VIMSS with the highest

**Table 2.1:** Three-fold cross-validation results for *E. coli* and *B. subtilis*. The area under the receiver operating curve (AUROC) is given for the three existing programs (JPOP, OFS, VIMSS) and different sets of inputs into the neural network. The number of inputs along with the different combinations of inputs is given from 3-fold cross-validation for each organism. The '3 only' represents the use of only the confidence scores of the three existing programs. The '3 + GO' represents the use of the three existing programs and GO similarity score for a total of 4 inputs into the neural network. Combinations using the pathway score and the intergenic distance scores are also given similarly. The neural network was fixed to be a simple 1-layer 1-neuron neural network with transfer function $f$=logsig. The majority of the improvement is realized by just combining the confidence scores from the three programs (3 only); however, there is further improvement by including other features such as GO similarity score, KEGG pathway score, and intergenic distance. The highest AUROC for each organism is shown in bold.

| 3-fold cross validation results | | | |
|---|---|---|---|
| # inputs | AUROC | *E. coli* | *B. subtilis* |
| - | JPOP | 0.8967 | 0.8568 |
| - | OFS | 0.9105 | 0.8381 |
| - | VIMSS | 0.9044 | 0.6207 |
| 3 | 3 only | 0.9225 | 0.8787 |
| 4 | 3 + GO | 0.9262 | 0.8797 |
| 4 | 3 + pathway | 0.9279 | 0.8798 |
| 4 | 3 + intergenic | 0.9170 | 0.8926 |
| 5 | 3 + GO + pathway | **0.9284** | 0.8802 |
| 5 | 3 + GO + intergenic | 0.9218 | 0.8955 |
| 5 | 3 + pathway + intergenic | 0.9267 | 0.8948 |
| 6 | 3 + GO + pathway + intergenic | 0.9275 | **0.8963** |

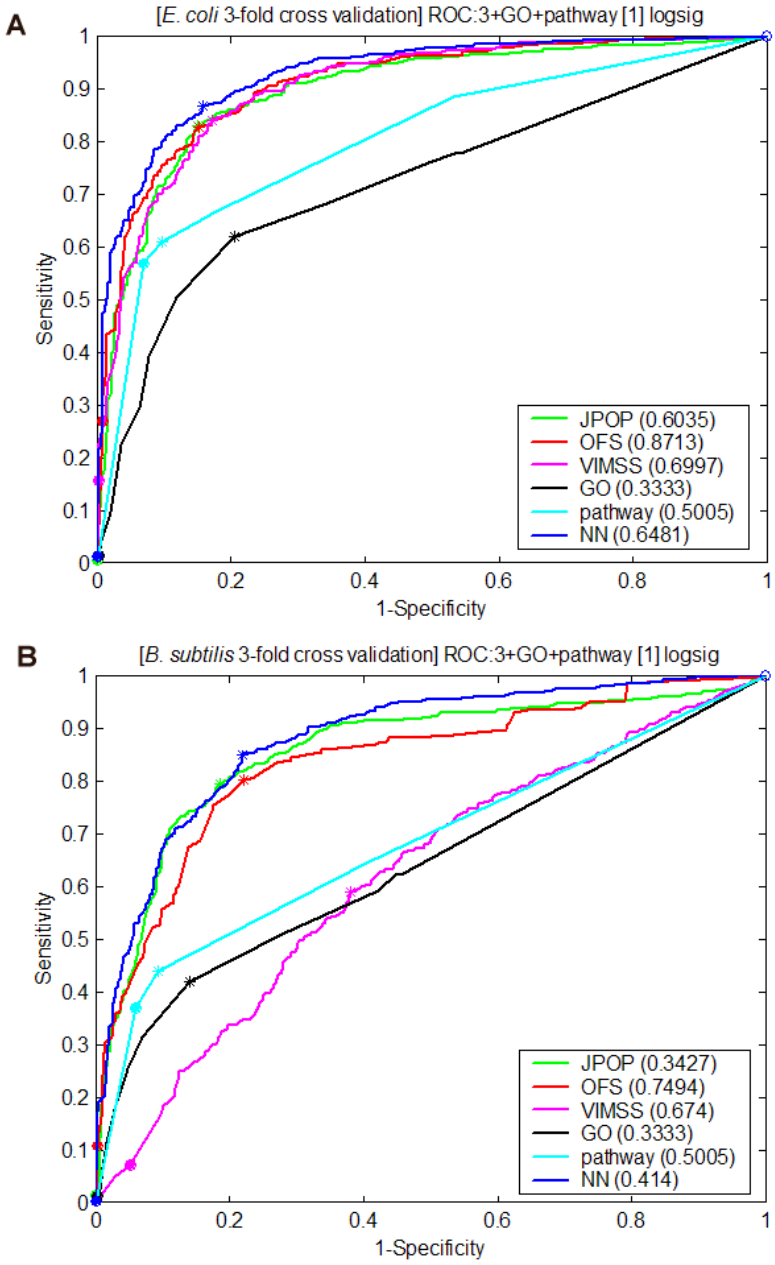


**Figure 2.7:** Three-fold cross validation results with 5 neural network inputs: 3 programs, GO similarity score, and KEGG pathway inputs for (A) *E. coli* and (B) *B. subtilis*. For all threshold levels, the neural network (NN) predictor is able to achieve higher Sensitivity and Specificity or comparable performance to the other existing operon prediction programs (JPOP, OFS, VIMSS). Each plot displays the ROC from the three existing programs {JPOP, OFS, VIMSS}, the performance of using only the GO similarity score {GO}, the performance of using only the pathway score {pathway}, and the performance of the neural network based predictor incorporating all of the aforementioned (5) features {NN}. The numbers in the legend correspond to the points indicated by an asterisk (*) in the plot showing each program's threshold that maximizes the (Sensitivity + Specificity) value.

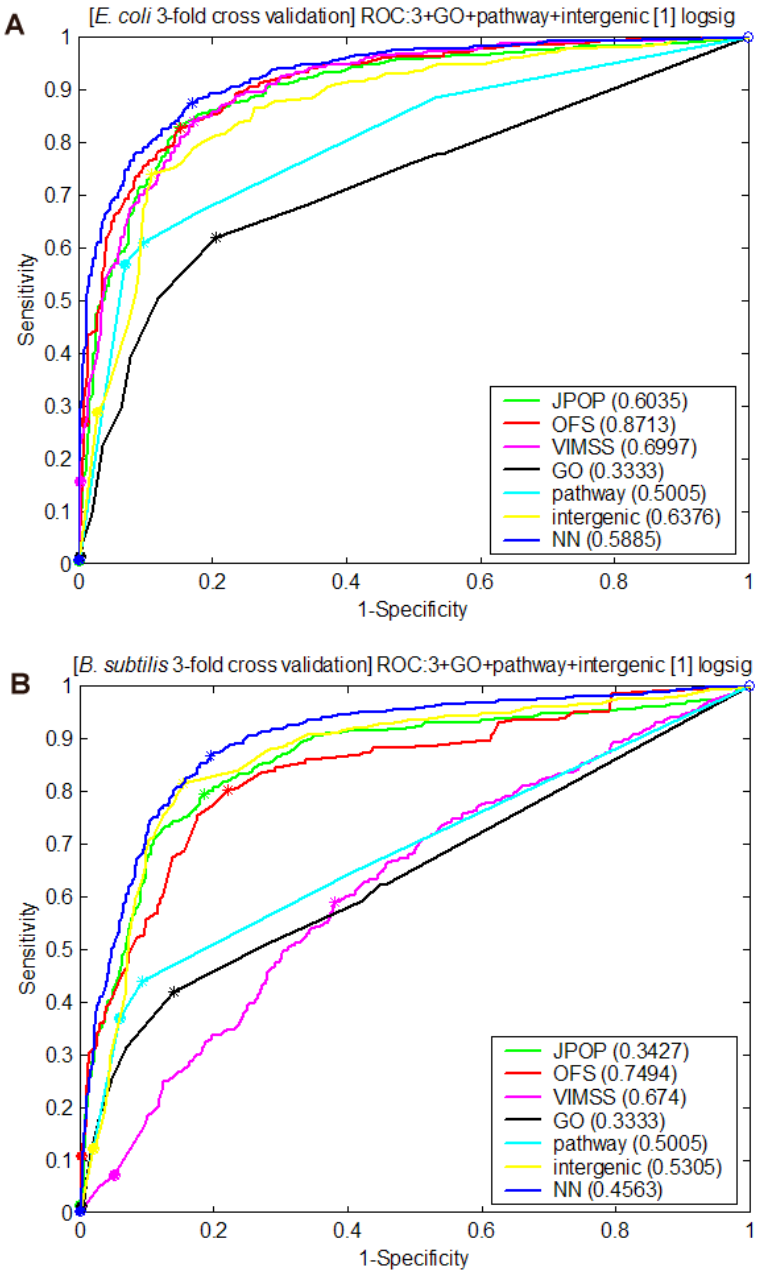


**Figure 2.8:** Three-fold cross validation results with 6 neural inputs: 3 programs, GO similarity score, KEGG pathway, and log-likelihood intergenic distance scores for (A) *E. coli* and (B) *B. subtilis*. For all threshold levels, the neural network (NN) predictor is able to achieve higher Sensitivity and Specificity than the other existing operon prediction programs (JPOP, OFS, VIMSS). Each plot displays the ROC from the three existing programs {JPOP, OFS, VIMSS}, the performance of using only the GO similarity score {GO}, the performance of using only the pathway score {pathway}, the performance of using only the log likelihood score of the intergenic distance {intergenic}, and the performance of the neural network based predictor incorporating all of the aforementioned (6) features {NN}. The numbers in the legend correspond to the points indicated by an asterisk (*) in the plot showing each program's threshold that maximizes the (Sensitivity + Specificity) value.

specificity among the three existing programs, the NN-based predictor was able to improve the specificity slightly and improve the sensitivity by almost 4%. In summary, the NN-based predictor was able to achieve higher sensitivity, specificity, and accuracy compared to the three existing programs. The graphical representation and parameters of this optimal network are shown in Figure 2.9 and Table 2.3. The corresponding ROC curves for the training and the testing sets are shown in Figure 2.10. The AUROC for our predictor on the *E. coli* test set is higher than any of the three other programs examined. We have also tested the scheme by reversing the training and testing data; however, the improvement to overall accuracy was marginal possibly because features in *B. subtilis* are more generalizable to other organisms. As a side note to Figure 2.10A, using only the intergenic distance in the *B. subtilis* training set seems to outperform the existing programs. As discussed in the previous section, this is not true in the case of *E. coli* so therefore additional features are still needed to aid prediction. Various factors have contributed to differences in performance among the programs, including (a) how other programs define their true positive/negative sets, (b) gene annotation data available at the time of prediction, (c) the prior distribution used in generating the intergenic distance scores, and (d) the contribution of the intergenic distance to each program's prediction. It is worth discussing why VIMSS performs poorly on the *B. subtilis* data set. We have investigated this by examining the histogram of the confidence scores for each program in Figures 2.11-2.13. The distribution of the VIMSS scores for *B. subtilis* seems to have poor discriminatory power between the TP and TN set. Of the reasons mentioned earlier, this is most likely due to the differences in the *B. subtilis* data sets used. Our training data set consisted of a list of 340 known operons published recently for *B. subtilis* versus the data set of 100 known operons as used in VIMSS. This and previously mentioned reasons could be investigated in later studies to help explain the poor performance of VIMSS on *B. subtilis* but good performance on *E. coli*.

### 2.4.3 Validation on *P. furiosus* prediction using *B. subtilis* training set

The parameters trained on *B. subtilis* were then applied to two data sets from *P. furiosus*: a "known operon" list consisting of 33 known/putative operons collected from the literature,

**Table 2.2:** Results of testing on *E. coli* after fixing network parameters and threshold from *B. subtilis*. The table presents the existing programs and various combinations of inputs into the NN predictor. The number in brackets [.] following each NN predictor indicates the number of neurons used in each layer. For example, [1] represents a single layer neuron with 1 neuron where [2,1] represents a two layer neuron network with two neurons in the hidden layer and 1 neuron in the output layer. For each program the following are given: the fixed threshold from *B. subtilis* training, sensitivity (Sn), specificity (Sp), and accuracy. In the *E. coli* testing set, there is improvement in overall accuracy, sensitivity, and specificity of the NN-based method over the existing three programs.

| | fixed | **Train (*B. subtilis*)** | | | **Test (*E. coli*)** | | |
|---|---|---|---|---|---|---|---|
| Program | threshold | Sn | Sp | Acc | Sn | Sp | Acc |
| JPOP | 0.3427 | 0.7962 | 0.8147 | 0.8049 | 0.8819 | 0.7433 | 0.8341 |
| OFS | 0.7494 | 0.8025 | 0.7788 | 0.7914 | 0.8819 | 0.7647 | 0.8415 |
| VIMSS | 0.6740 | 0.5892 | 0.6187 | 0.6030 | 0.8453 | 0.8182 | 0.8359 |
| NN (3+GO+pathway)[1] | 0.4164 | 0.8519 | 0.7788 | 0.8176 | 0.9241 | 0.7273 | 0.8562 |
| NN (3+GO+pathway+intergenic)[1] | 0.4756 | 0.8662 | 0.8219 | 0.8454 | 0.8903 | 0.7861 | 0.8544 |
| NN (3+GO+pathway+intergenic)[2,1] | 0.5876 | 0.8328 | 0.8651 | 0.8480 | 0.8847 | 0.8262 | 0.8645 |

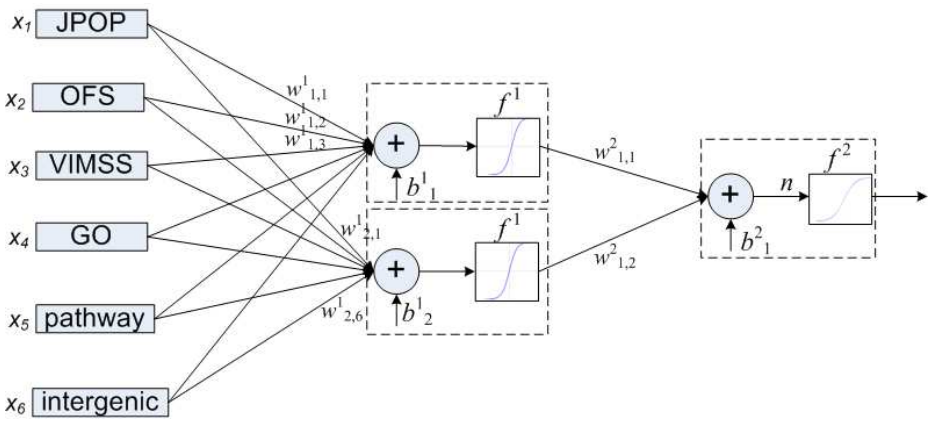


**Figure 2.9:** Two-layer neural network architecture used in the training/test set validation. The inputs to the network are confidence scores from each operon prediction program and additional features from GO similarity, KEGG pathway, and intergenic distance scores $\{x_i\}$. The first-layer (hidden layer) consists of two neurons with transfer function $f^1$. The second-layer (output layer) consists of one neuron with transfer function $f^2$. The superscripts indicate the layer number of each parameter. Weights are denoted by $w^{<layer\#>}_{<destination,source>}$ and biases are denoted by $b^{<layer\#>}_{<neuron\#>}$.

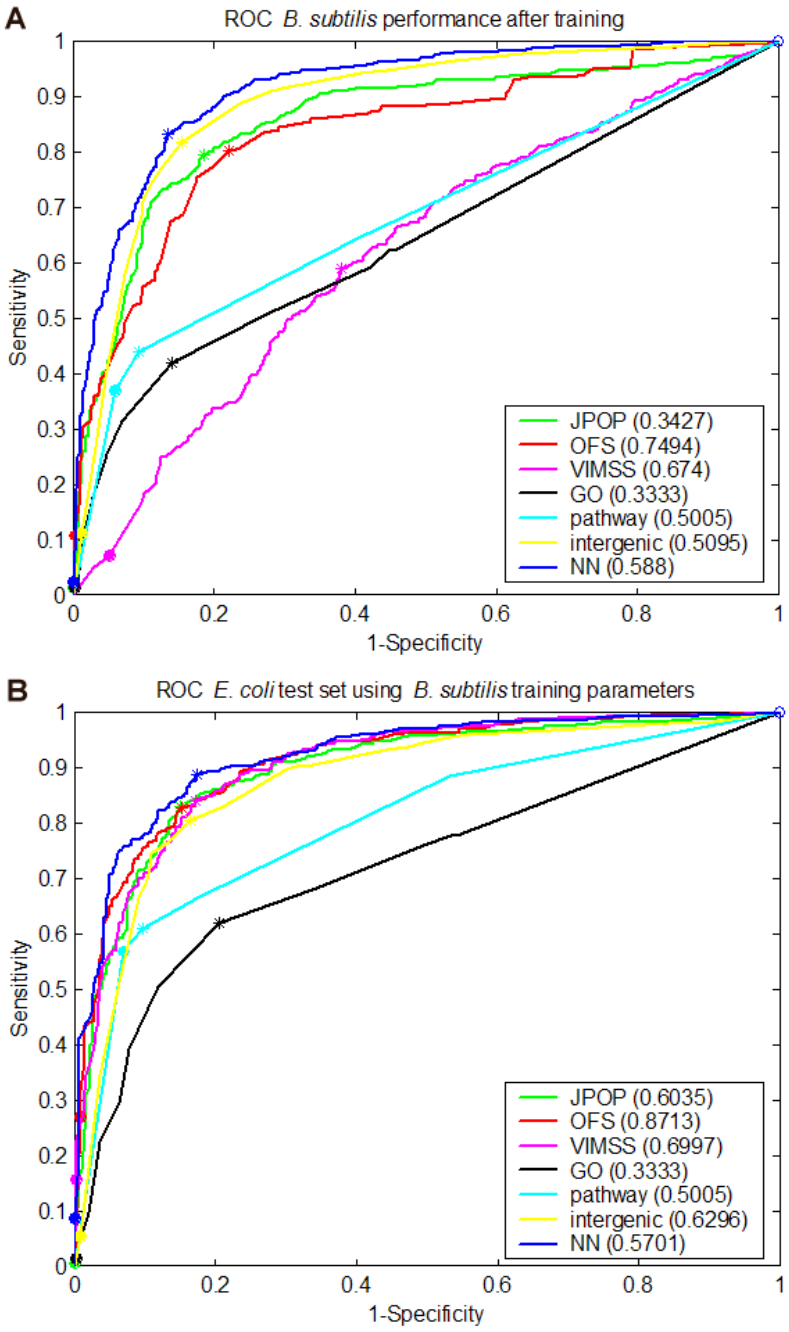


**Figure 2.10:** (A) ROC for the *B. subtilis* training set. (B) ROC for *E. coli* using trained parameters from *B. subtilis*. Each plot displays the ROC from the three existing programs {JPOP, OFS, VIMSS}, the performance of using only the GO similarity score {GO}, the performance of using only the pathway score {pathway}, the performance of using only the log likelihood score of the intergenic distance {intergenic}, and the performance of the neural network based predictor incorporating all of the aforementioned (6) features {NN}. The numbers in the legend correspond to the points indicated by an asterisk (*) in the plot showing each program's threshold that maximizes the (Sensitivity+Specificity) value. For any threshold, the NN-based method has higher performance than any of the existing programs.

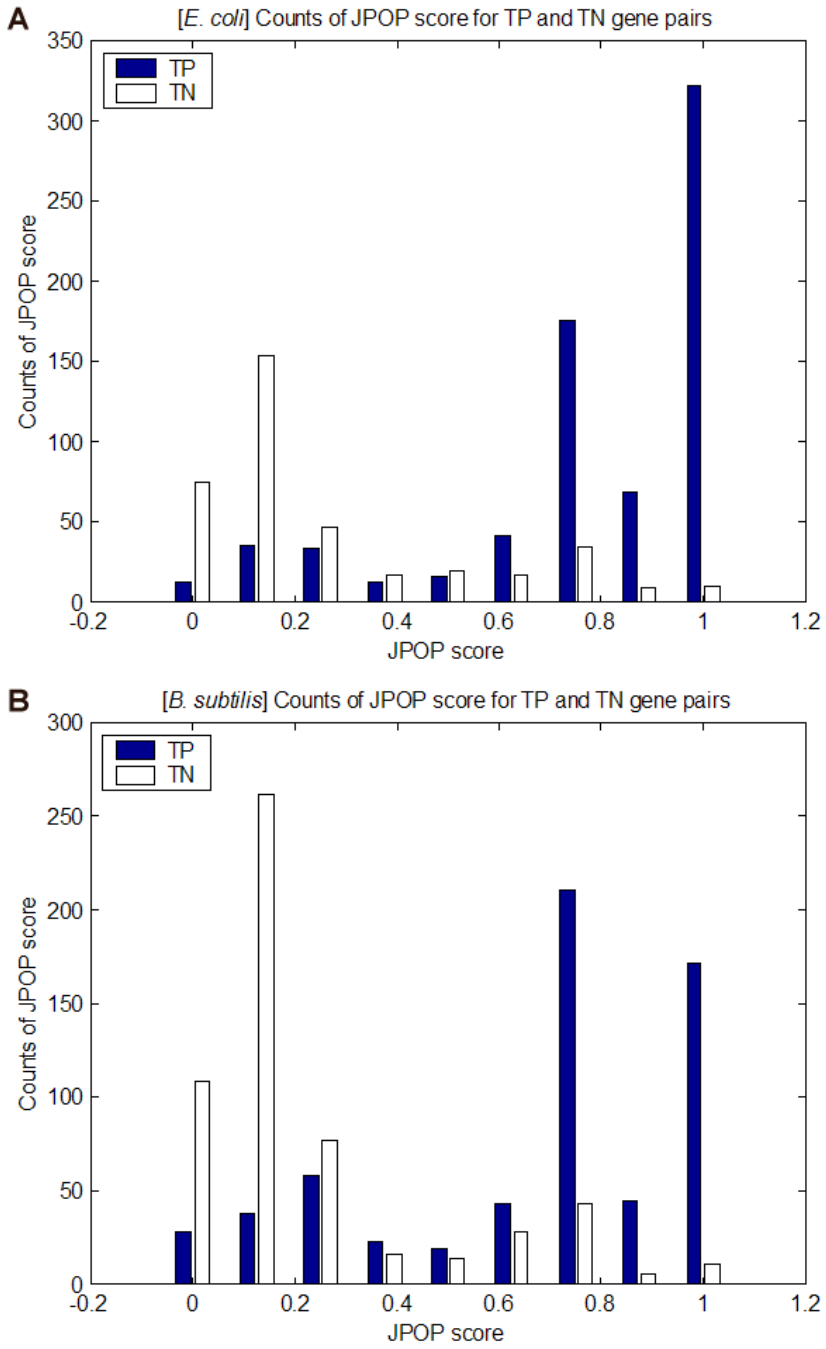


**Figure 2.11:** Histogram of JPOP confidence scores. The counts of the confidence scores for the true positive (TP) and true negative (TN) set are given for (A) *E. coli* and (B) *B. subtilis*. The histogram for each organism shows the TP data clustering around high JPOP confidence scores and the TN data clustering around low JPOP confidence scores.

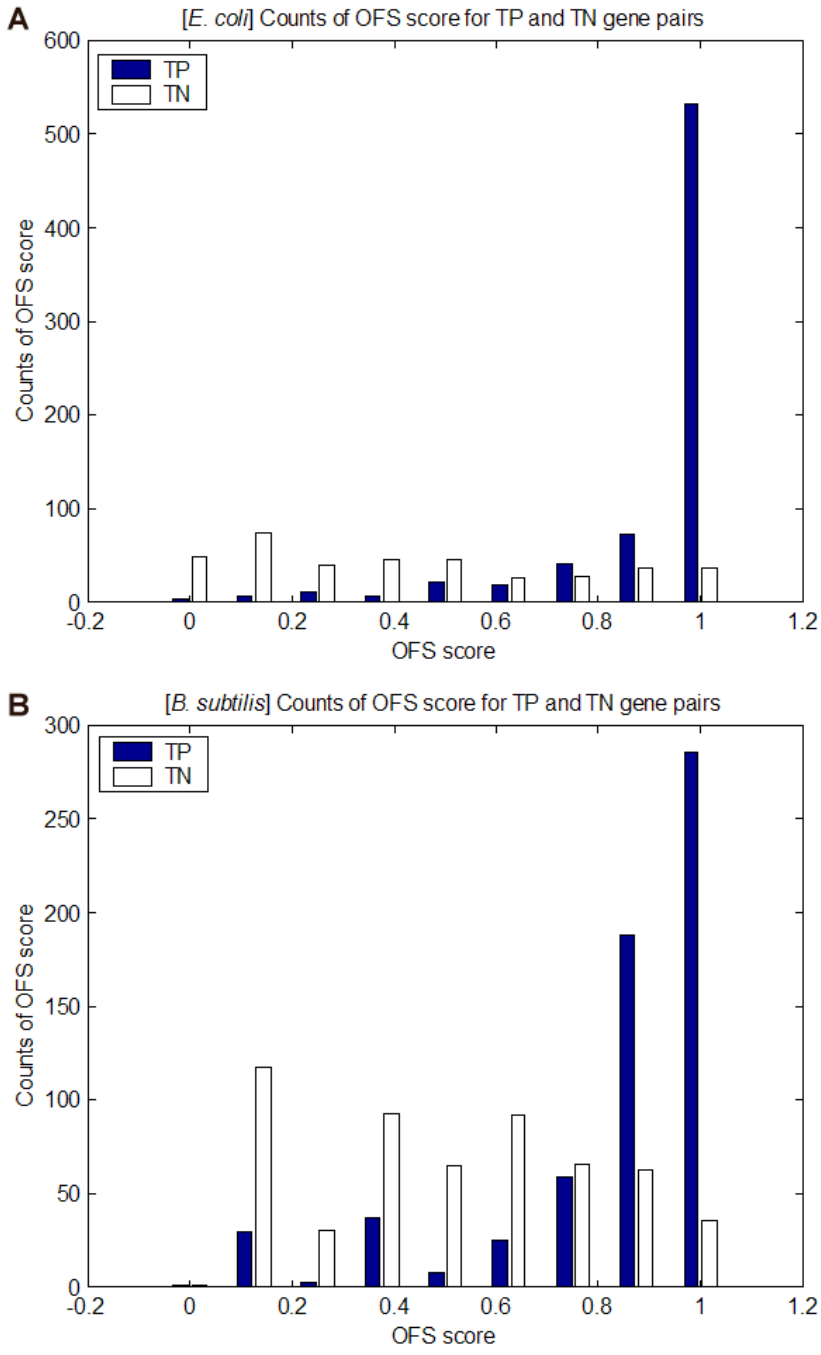


**Figure 2.12:** Histogram of OFS confidence scores. The counts of the confidence scores for the true positive (TP) and true negative (TN) set are given for (A) *E. coli* and (B) *B. subtilis*. The histogram for each organism shows the TP data clustering around high OFS confidence scores while the distribution for the TN set is more uniform.

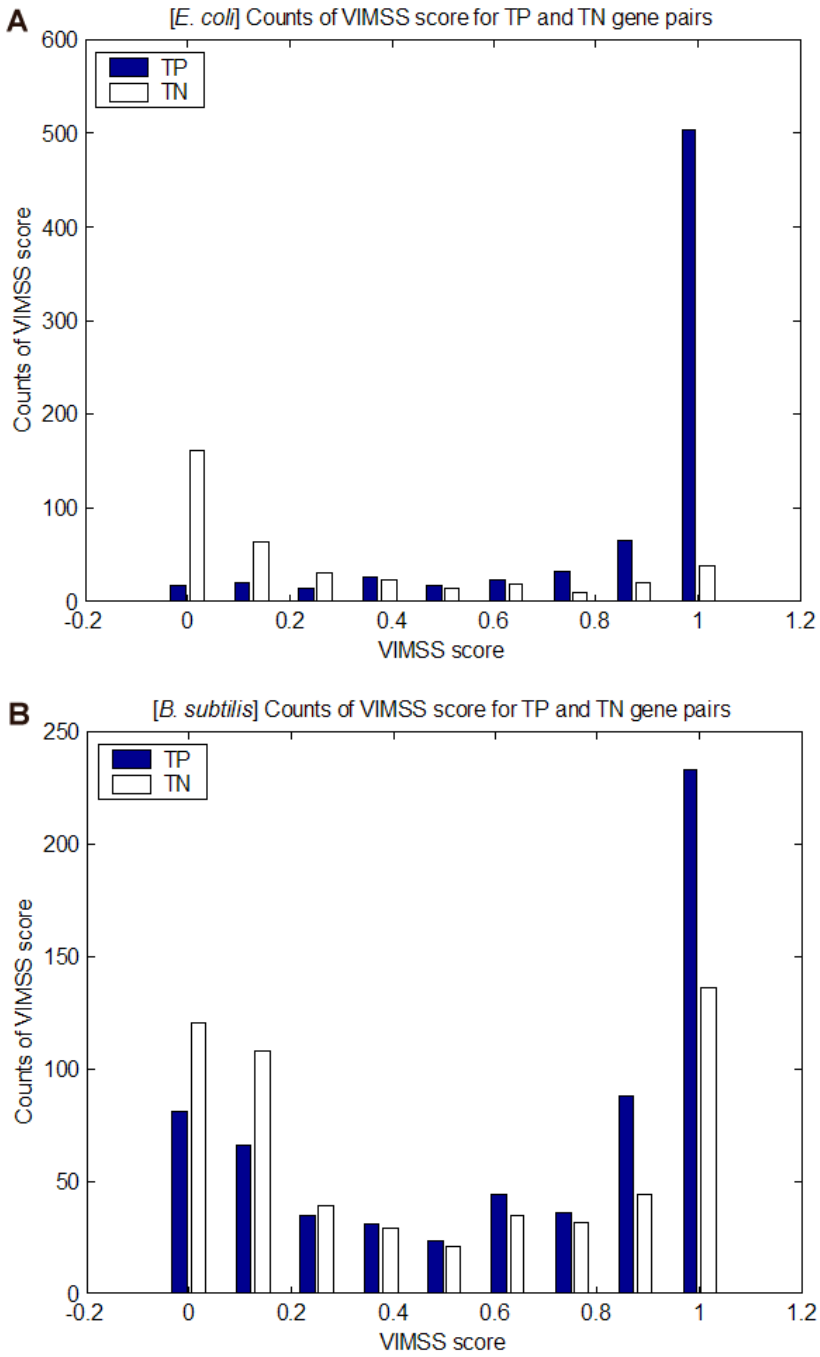


**Figure 2.13:** Histogram of VIMSS confidence scores. The counts of the confidence scores for the true positive (TP) and true negative (TN) set are given for (A) *E. coli* and (B) *B. subtilis*. The histogram for *E. coli* shows the TP data clustering around high VIMSS confidence scores while the distribution for the TN data clusters around lower scores. However, in the *B. subtilis* data set, the histogram of the TP and the TN set is bimodal and quite similar. This indicates lower performance of VIMSS for *B. subtilis* in separating the TP and TN set as used in this study.

**Table 2.3:** Optimal two-layer neural network parameters used in the training/test set validation. The notation in the table can be found in Figure 2.9.

| Hidden layer | | Output layer | |
|---|---|---|---|
| $w^1_{1,1}$ | 430.3567 | $w^2_{1,1}$ | 1.2248 |
| $w^1_{1,2}$ | 13.0787 | $w^2_{1,2}$ | 4.1703 |
| $w^1_{1,3}$ | 119.6073 | | |
| $w^1_{1,4}$ | 1154.5 | | |
| $w^1_{1,5}$ | -296.3242 | | |
| $w^1_{1,6}$ | 148.9444 | | |
| $w^1_{2,1}$ | 0.0441 | | |
| $w^1_{2,2}$ | 0.2723 | | |
| $w^1_{2,3}$ | 0.0935 | | |
| $w^1_{2,4}$ | -0.3106 | | |
| $w^1_{2,5}$ | 0.3515 | | |
| $w^1_{2,6}$ | 1.2330 | | |
| $b^1_1$ | -827.8975 | $b^2_1$ | -0.1447 |
| $b^1_2$ | -0.6022 | | |
| $f^1$ | tansig | $f^2$ | logsig |

and the "microarray evidence list" as described in the Methods section. The testing results on these two sets are shown in Table 2.4. It is interesting to note that the prediction sensitivity is much higher at the expense of specificity. This is probably an intrinsic limitation of applying trained parameters from one species to another (more distant) species. The optimal performance for the "known operon" list can be found in Figure 2.14 and Table 2.5. From Table 2.5, using the optimal *P. furiosus* threshold, the NN-based prediction has highest accuracy compared to any of the three methods. Depending on the tradeoff between the sensitivity and the specificity, a user can best decide which threshold is best for their specific application. Whichever way the threshold is chosen in *P. furiosus* (whether the default training threshold from *B. subtilis* or the optimal threshold), the prediction sensitivity is consistently higher at the expense of lower specificity. A similar trend was observed with the "microarray evidence list" over a range of threshold values as shown in Figure 2.15. Although the NN-based method achieves the primary goal of our study in having higher overall accuracy compared to other programs, the results tend to over-predict operons and give a higher number of false positives. Between sensitivity and specificity; however, we would prefer a higher sensitivity so as not to miss many operons. Since the objective of our computational prediction is to provide a list of potential operons in *P. furiosus* so that further experimental validation can be performed, this prediction is acceptable in our case since we have microarray data to further filter out the false predictions. It should be noted

**Table 2.4:** Results of testing on *P. furiosus* using fixed network parameters and threshold from *B. subtilis*. The results are from applying an optimal 2-layer (2-neuron hidden layer with a tansig transfer function and a 1 output neuron with a logsig transfer function) neural network. The NN-based method presented uses inputs from the three existing programs together with GO, pathway, and intergenic scores. The sensitivity (Sn), specificity (Sp), and accuracy are given for each program under each test set. "Known operons" is a limited set of 33 known/putative operons from literature. The microarray evidence list is described in the microarray data analysis section.

| [2,1] tansig logsig | fixed | known operons | | | microarray evidence list | | |
|---|---|---|---|---|---|---|---|
| Program | threshold | Sn | Sp | Accuracy | Sn | Sp | Accuracy |
| JPOP | 0.3427 | 0.8972 | 0.6129 | 0.8333 | 0.8198 | 0.5657 | 0.7453 |
| OFS | 0.7494 | 0.8505 | 0.7419 | 0.8261 | 0.5545 | 0.6936 | 0.5953 |
| VIMSS | 0.6740 | 0.8972 | 0.7097 | 0.8551 | 0.7249 | 0.6970 | 0.7167 |
| NN | 0.5876 | 0.9907 | 0.5806 | 0.8986 | 0.9022 | 0.5354 | 0.7947 |

**Table 2.5:** Results of applying the "known operons" data set of *P. furiosus* at the optimal threshold. The results are from applying an optimal 2-layer (2-neuron hidden layer, 1 output neuron) neural network trained on *B. subtilis*. For each program the following are given: the optimal threshold from testing, sensitivity (Sn), specificity (Sp), and accuracy.

| [2,1] tansig logsig | optimal | known operons | | |
|---|---|---|---|---|
| Program | threshold | Sn | Sp | Accuracy |
| JPOP | 0.6299 | 0.8692 | 0.7097 | 0.8333 |
| OFS | 0.6793 | 0.9065 | 0.7097 | 0.8623 |
| VIMSS | 0.8683 | 0.7944 | **0.9032** | 0.8188 |
| NN | 0.7293 | **0.9533** | 0.7097 | **0.8986** |

that these two test sets in *P. furiosus* are by no means complete and this initial attempt to assess the performance of the NN-based method on a new organism is a rough indicator for what one can expect based on this limited data.

### 2.4.4 Whole-genome operon prediction

The parameters trained on the *B. subtilis* set were applied to the entire genomes of *E. coli*, *B. subtilis*, and *P. furiosus*. A summary of the predicted operons for *E. coli*, *B. subtilis*, and *P. furiosus* is shown in Table 2.6. The NN-based method predicted 470 operons covering 1,460 ORFs in *P. furiosus*. An average operon consists of 3.1 ORFs. A summary of the number of operons predicted for each organism along with other statistics for the different programs is given in Table 2.7.

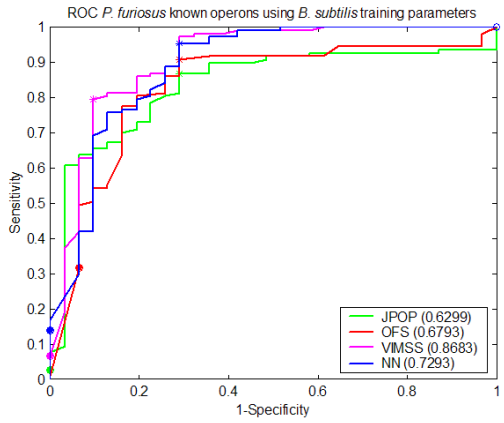


**Figure 2.14:** ROC curve for the "known operon" test set from *P. furiosus*, which consists of 33 known/putative operons from literature. The results use an optimal 2-layer (2-neuron hidden layer, 1 output neuron) NN trained on *B. subtilis*. The plot displays the ROC from the three existing programs {JPOP, OFS, VIMSS} and the performance of the neural network based predictor using 6 (JPOP, OFS, VIMSS, GO similarity, pathway, intergenic distance) features {NN}. The values in the legend correspond to the points indicated by an asterisk (*) in the plot showing each program's threshold that maximizes the (Sn + Sp) value. The overall accuracy at this optimum threshold is highest in the NN method compared to any of the other programs. The actual values are computed in Table 2.5.

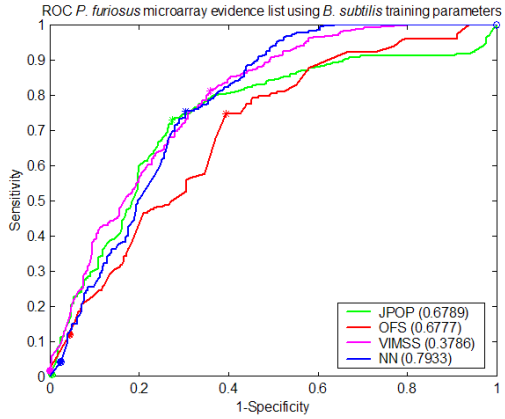


**Figure 2.15:** ROC curve for the "microarray evidence list" test set from *P. furiosus*, as discussed in Section 2.3. The results are from applying an optimal 2-layer (2-neuron hidden layer, 1 output neuron) neural network trained on *B. subtilis*. The plot displays the ROC from the three existing programs {JPOP, OFS, VIMSS} and the performance of the neural network based predictor using 6 (JPOP, OFS, VIMSS, GO similarity, pathway, intergenic distance) features {NN}. The values in the legend correspond to the points indicated by an asterisk (*) in the plot showing each program's threshold that maximizes the (Sn + Sp) value. Over a range of threshold values, the Sn of the NN-based method is higher than the other programs at the expense of Sp. With improved GO and pathway annotation in *P. furiosus*, it is expected that the performance of the NN-based method will improve over other methods at other thresholds.

**Table 2.6:** Characteristics of operons predicted by the NN-based method for each organism. For each organism, the number of open reading frames (ORFs) included in the operon prediction, the number of operons, the average operon size, and the percent of gene coverage (=100*#ORFs included in the operon prediction/Total #ORFs in the organism) are given.

| Organism | # ORFs | # operons | Ave op size | % gene coverage |
|---|---|---|---|---|
| *E. coli* | 2490 | 806 | 3.0893 | 59% |
| *B. subtilis* | 2288 | 747 | 3.0629 | 56% |
| *P. furiosus* | 1460 | 470 | 3.1064 | 69% |

**Table 2.7:** Summary of the predicted operons from each program at the optimal *B. subtilis* training threshold. For each program and organism, the number of open reading frames (ORFs) included in the operon prediction, the number of operons, the average operon size, the percent of gene coverage (=100*#ORFs included in the operon prediction/Total #ORFs in the organism), and the number of gene pairs included in the operon prediction are given.

| NN method threshold = 0.5876 | # ORFs | # operons | ave op size | % gene coverage | # gene pairs |
|---|---|---|---|---|---|
| *E. coli* | 2490 | 806 | 3.0893 | 59% | 1684 |
| *B. subtilis* | 2288 | 747 | 3.0629 | 56% | 1541 |
| *P. furiosus* | 1460 | 470 | 3.1064 | 69% | 990 |
| JPOP threshold = 0.3427 | # ORFs | # operons | ave op size | % gene coverage | # gene pairs |
| *E. coli* | 2164 | 759 | 2.8511 | 51% | 1405 |
| *B. subtilis* | 2294 | 778 | 2.9486 | 56% | 1516 |
| *P. furiosus* | 1389 | 476 | 2.9181 | 65% | 913 |
| OFS threshold = 0.7494 | # ORFs | # operons | ave op size | % gene coverage | # gene pairs |
| *E. coli* | 2147 | 663 | 3.2383 | 51% | 1484 |
| *B. subtilis* | 2350 | 774 | 3.0362 | 57% | 1576 |
| *P. furiosus* | 904 | 305 | 2.9639 | 43% | 599 |
| VIMSS threshold = 0.674 | # ORFs | # operons | ave op size | % gene coverage | # gene pairs |
| *E. coli* | 2284 | 763 | 2.9934 | 54% | 1521 |
| *B. subtilis* | 1640 | 551 | 2.9764 | 40% | 1089 |
| *P. furiosus* | 1118 | 394 | 2.8376 | 53% | 724 |

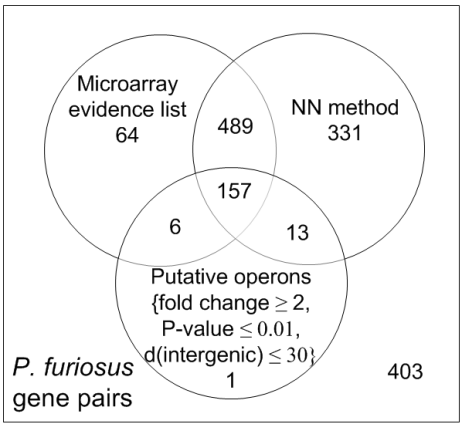


**Figure 2.16:** Venn diagram of overlap between gene pairs for operons predicted from the NN-based method, the "microarray evidence list", and the "putative operon list". Predicted operons from the NN-based method overlapping the "microarray evidence list" and the "putative operon list" represent strong candidates for further experimental studies.

### 2.4.5 Functional annotation of *P. furiosus* operons

The predicted operons that overlap the "microarray evidence list" are annotated and can be found at http://csbl.bmb.uga.edu/∼tran/operons. The annotation for *P. furiosus* was obtained from Genbank and the TIGR-Comprehensive Microbial Resource [117]. In addition, a subset of this list that overlaps the "putative operon list" can be found at http://csbl.bmb.uga.edu/ tran/operons. The number of overlapping gene pairs from these files are summarized in the Venn diagram as shown in Figure 2.16. The 646 (=489+157) gene pairs common to our predicted operons and the "microarray evidence list" represent 349 unique operons. The 157 gene pairs that overlap all three lists form 98 operons. The novel operons in this set provide biologists a list of targets for further experimental studies.

The 71 (=64+6+1) gene pairs in Figure 2.16, which were not predicted by the NN-based method may be due to a combination of microarray experimental errors, the methodology used in microarray analysis, or prediction errors. The "putative operon list" has higher specificity than the "microarray evidence list" in terms of applying a lower intergenic distance cutoff and more microarray conditions, which can help to reduce the number of false predictions. As a result, the majority of the gene pairs from the "putative operon list" overlap with the NN-based prediction. The gene pairs in the microarray evidence list not predicted by the NN-based method represent a small fraction of total gene pairs in the list

(∼10%). These gene pairs may be errors in predictions or could be due to noisy microarray data. The 331 gene pairs predicted by the NN-based method but not present in either of the microarray lists may be due to over-prediction or because the condition for co-expression may not have been tested in the microarray data available.

Our NN-based method was able to detect both of the two operons in *P. furiosus* that are previously known, namely the lamA and POR/VOR operons. In the case of the POR/VOR operon, which consists of 3 mRNA transcripts: [PF0965 PF0966 PF0967] [PF0968 PF0969 PF0970] [PF0971], our method predicted these as one large operon from PF0965 to PF0971. This is attributed to the fact that all three existing programs predict the POR/VOR operon as one single transcript. This is an intrinsic limitation in combining existing methods though we expect that such cases are rare.

### 2.4.6 Summary

Operon prediction allows for the functional inference of hypothetical and conserved hypothetical genes, and represents a key step in reconstructing biological pathways and networks for prokaryotes. A novel neural network-based approach for operon prediction is described herein that integrates the strengths of existing prediction algorithms, which use various sequence features such as codon usage and intergenic distance, conserved gene order, phylogenetic profiles of genes, and COG functional annotation. By integrating the prediction results of the three programs, we are able to achieve better performance by taking advantage of the complementary information provided by each individual program. By using GO annotation, KEGG pathway, and intergenic distance information as additional inputs into our program, we have further improved upon the accuracy. The improvement in performance by our new algorithm is demonstrated through cross-validation on *E. coli* and *B. subtilis*, and also through the test results on *E. coli* using *B. subtilis* data as the training set.

The use of GO annotation and KEGG pathway only improves the NN-based prediction results slightly. This is partially due to the low coverage of GO and KO annotation in each of the involved organisms. Even with a well-studied organism like *E. coli*, the coverage

is only 50% and 61% for GO and KO annotation, respectively. With improved genome annotation, it is expected that the application of such information should provide a higher level of improvement in the NN-based method. One major limitation of our method, which is typical of most machine learning approaches, is that it requires the use of training data. In our case, the parameters of the NN and the optimal threshold were fixed based on *B. subtilis* data, and applied to other prokaryotic organisms. This is a general problem with existing operon prediction algorithms which use features that could be species dependent. For example, the intergenic distance to distinguish between operon gene pairs and non-operon gene pairs can vary substantially depending on species.

For our future work, other machine learning methods such as support vector machines and decision trees can also be investigated as alternative approaches to improve the prediction results. Furthermore, the NN-based prediction method can be expanded to include newer operon prediction programs as they become available. Further analysis to reduce the dimension of the input feature space could also be performed. This could involve testing a neural network architecture based on only one or two existing programs in conjunction with combinations of the GO similarity, KEGG pathway, and intergenic distance score. With a list of predicted operons in *P. furiosus*, further study can be done to identify potential regulons (groups of operons sharing a common regulatory mechanism). For each operon, we can examine the region approximately 250 bp upstream and use prediction algorithms such as CUBIC [110] or MEME [10] to predict potential binding sites for transcriptional factors, and use the shared binding motifs as initial indicators for potential regulons.

The operon prediction algorithm presented in this paper coupled with GO, KEGG, and microarray analysis brings forth the most comprehensive prediction of operon structure in the organism *P. furiosus* to date. This approach can similarly be applied to other prokaryotic organisms with complete genomes. The computationally predicted operons for *P. furiosus* in this study paves the way for further computational and experimental investigation into a better understanding of the regulatory pathway of this hyperthermophilic archaeon.

# CHAPTER III

# *DE NOVO* COMPUTATIONAL PREDICTION OF NON-CODING RNA GENES IN PROKARYOTIC GENOMES[1]

The computational identification of non-coding RNA (ncRNA) genes represents one of the most important and challenging problems in computational biology. Existing methods for ncRNA gene prediction rely mostly on homology information, thus limiting their applications to ncRNA genes with known homologues. We present a novel *de novo* prediction algorithm for ncRNA genes using features derived from the sequence and structures of known ncRNA genes in comparison to decoy sequences. Using these features, we have trained a neural network-based classifier and have applied it to *Escherichia coli* for genome-wide prediction of ncRNAs. Our method has an average prediction sensitivity and specificity of 68% and 70%, respectively, for identifying windows with potential for ncRNA genes. By combining windows of different sizes, we can recover 84/93 known ncRNA genes in *E. coli*. However, this approach results in a relatively high false positive rate, which can be reduced through additional filtering strategies. We performed Northern blot analysis on six candidates and found expression of three candidates, which may be stable decay intermediates or in one case a potential riboswitch. Our approach enables the identification of both cis- and trans- acting ncRNAs in partially or completely sequenced microbial genomes without requiring homology or structural conservation.

## *3.1 Introduction*

Non-coding RNA (ncRNA) or small RNA (sRNA) genes, which encode functional RNA molecules that are not translated into proteins, are involved in a variety of cellular processes ranging from regulation of gene expression to RNA modification and editing [58, 66]. In humans, it is estimated that about 98% of the genome can be transcribed, of which only

[1]This chapter is a result of joint work with Fengfeng Zhou, Sarah Marshburn, Mark Stead, Sidney R. Kushner, and Ying Xu.

$\sim$2% encodes protein genes [140], suggesting the possibility that a large percentage of the genome may encode ncRNA genes. It is believed that cellular regulation by ncRNAs account for much of an organism's complexity, particularly for higher-level organisms [102, 103]. Although the vital importance of ncRNA genes in cellular activities is well recognized, our current knowledge about the collection of all ncRNA genes encoded in a particular genome is very limited because of the lack of effective capabilities, either computational or experimental, for elucidating them.

It is generally believed that the identification of ncRNA genes, particularly in prokaryotic genomes, is more challenging than protein-coding genes. Unlike protein-coding genes, ncRNA genes do not contain easily detectable signals such as open reading frames (i.e., a sequence between an in-frame start codon and the first in-frame stop codon going from the 5' to the 3' end of the sequence), codon biases, or ribosome binding sites (RBS). Although some ncRNA genes have recognizable promoters and terminators [6, 21], the identification of such regulatory signals is quite challenging [40]. This identification problem is further complicated by the fact that most ncRNA genes are much shorter than protein-coding genes [4].

A number of computational methods for identifying ncRNA genes have been developed and reported [6, 158, 97, 170, 173, 118, 156, 21, 165, 83, 81, 132]. These methods generally fall into two classes: (1) methods that identify members of an ncRNA family based on homology information and (2) methods that find novel ncRNAs based on general features common to ncRNA genes.

(1) *Homology-based methods for ncRNA gene prediction*: Homology-based methods have been widely used for the prediction of ncRNA genes with known sequence or structural homologues by using BLAST or more sophisticated search techniques [6, 158, 53]. While effective for identifying numerous ncRNA genes across different genomes [97], they are not designed to find novel ncRNA genes. Compared to protein-encoding genes, additional challenges hinder the identification of homologous RNA-encoding genes since (a) ncRNAs are generally much shorter compared to protein-coding genes and (b) ncRNA sequences appear to be significantly less conserved than protein-encoding sequences [31, 146, 81, 114].

To overcome these issues, the secondary structures of ncRNA genes, which are more conserved than sequences, have often been used to complement sequence-based homology search. One popular class of prediction methods employs stochastic context-free grammars (SCFG) to incorporate secondary structure information into the search for homologous ncRNAs [170]. By combining aligned sequences from a specific class of ncRNAs and their predicted secondary structures, a SCFG model can be generated and used to search a database for new candidates [59, 80, 56, 169].

(2) *De novo methods for novel ncRNA gene prediction*: Two methods have been developed to predict novel ncRNAs. The first method identifies (relatively long) conserved sequences in the intergenic regions across closely related genomes. This method is based on the assumption, which is generally true for prokaryotic genomes, that such conserved regions encode functional trans-acting ncRNAs and not cis-regulatory motifs. Such a strategy has used to mine *E. coli* [173] and other bacterial organisms [118] for novel ncRNAs. By limiting the search to intergenic regions, one could realistically search for ncRNA genes on a genome-wide scale. However, this approach will miss ncRNAs that overlap protein-coding genes, either cis- or trans-, and ncRNA genes that are unique to a genome. For example, it is known that ∼25% of the C/D snoRNA genes overlap protein-coding genes in the *Pyrococcus abyssi* genome [55]. A generalization of this type of method is to predict novel ncRNA genes through the identification of conserved RNA secondary structures across related genomes and further analyze their mutational patterns [122, 123] or evaluate for the folding energy of the predicted structures [26, 32, 115, 156]. However, such structure-based methods may suffer from having low prediction reliability [156].

The second method predicts novel ncRNA genes based on identifying both common and distinguishing features of known ncRNA genes in target genomic regions. The features used have included predicted promoters and terminators, as well as the base compositions of target sequences. Typical requirements mandate that such a region be short and flanked by promoter and terminator signals [6, 21, 165, 83]. Clearly, such methods are limited in their effectiveness to reliably predict novel ncRNA genes for two main reasons: (a) accurate prediction of such signals is very challenging and unreliable and (b) only a fraction of

terminators, namely, rho-independent terminators in prokaryotes, can be computationally predicted [79].

Although nucleotide composition-based methods have had some success in ncRNA gene prediction, these methods are limited to organisms with compositional bias in their ncRNA genes in relation to their underlying genome. For example, in A/T-rich hyperthermophilic genomes, the ncRNA genes are in general highly G/C rich [81, 132, 87]. In addition to base composition (or mono-nucleotide composition), some programs have employed di- and tri-nucleotide frequencies to distinguish ncRNA genes from the genomic background [153]. Such information has also been further enhanced through the use of folding energy and known RNA motifs [18] for the prediction of ncRNA genes in *E. coli*.

In this paper, we present a *de novo* method for predicting ncRNAs in prokaryotic genomes, using a number of novel structural features associated with known ncRNA genes. A neural network-based classifier was trained to predict the ncRNA genes on a genome-wide scale. We have applied this classifier to RNA gene prediction in *E. coli* and have compared our predictions to other existing programs. Furthermore, we also experimentally investigated six of the novel candidates identified by the algorithm using Northern blot analysis and identified a potential riboswitch that may help regulate the expression of the mreB gene.

## *3.2 Materials and Methods*

To train a classifier for the *de novo* prediction of ncRNAs genes, we first generated a positive data set containing known ncRNA genes and identified a set of sequence and structural-based features that could distinguish the positive data set from non-ncRNA genes. We assume that ncRNA genes are no longer than 1,000 nucleotides (nt), which covers the vast majority of the known ncRNAs in prokaryotes.

### 3.2.1 Data set generation

Our positive ncRNA data set was derived from three existing sources: (1) the NONCODE database [93], (2) published literature, and (3) GenBank. We did not consider tRNAs and rRNAs in our positive data set, since they can be identified using current methods [98, 84].

Non-coding RNA sequences from the NONCODE database were downloaded and filtered to extract only those belonging to prokaryotic organisms. We used BLASTN to find the exact position of each ncRNA in its host genome. Our literature search yielded ncRNA gene annotations for four prokaryotic organisms [141, 142, 78, 111, 172]. Additionally, the NCBI RNA annotation was searched for entries containing "RNA" but not "tRNA" or "rRNA." These searches yielded 427, 426, and 1,105 ncRNAs from NONCODE, published literature, and NCBI, respectively, for a total of 1,540 non-overlapping ncRNAs, which we refer to as "Positive1540" for future reference.

To remove redundant sequences within this data set, we applied the Markov cluster (MCL) algorithm to group together similar sequences using the default inflation parameter and a BLAST bit-score cutoff of 5 [45]. Application of this algorithm resulted in 936 clusters from which we randomly selected one ncRNA from each cluster to use in our final training data set. We refer to this data set as "Positive936" to represent our known positive controls.

The generation of the negative control represented a challenge in our work since there are no known negative sets, i.e., regions of the genome known not to contain ncRNA genes. Approaches using segments of the genomic background [18, 132, 129] as the control inherently assume that ncRNA genes make up only a small portion of the entire genome, which may not be correct. Other methods use randomly shuffled permutations of known ncRNA genes to build a negative training data set [25, 81, 162, 121]. We constructed our negative set by shuffling sequences of known ncRNA genes, while preserving both the mono- and di-nucleotide frequencies. This approach prevented the negative set from being biased to certain regions of the genome. The rationale for preserving the compositional frequencies was that it enabled the calculation of the minimum folding energy (MFE) without biasing the stabilizing and destabilizing energy from stacked base pairs or loops, respectively [162, 25, 52]. We used the shuffling strategy implemented in Clote et al. [25], based on the Altschul and Erickson algorithm [3]. We use the term "di-shuffle" to represent the shuffling procedure that preserves the mono- and di-nucleotide frequencies. For each known ncRNA sequence in "Positive936," we generated 1,000 di-shuffled sequences, to which we refer as "Dishuffle936."

### 3.2.2 Features used

Secondary structures play a key role in the functions of ncRNAs and are more highly conserved than the primary sequences. Accordingly, we investigated a number of secondary structure-based features in terms of their power to differentiate between ncRNAs and their di-shuffled sequences, including novel features such as structural and ensemble statistics, plus a few previously used features such as folding statistics.

#### *3.2.2.1 Folding statistics*

We examined the MFE [156, 18, 153] distributions for real ncRNAs and their di-shuffled sequences. Although useful, the current thermodynamic model used in RNA secondary structure prediction is accurate to only within 5-10% of the actual MFE, making the accuracy of the current MFE-based structure predictions around 50-70% [42]. Therefore we used other features in conjunction with MFE to assess the reliability of the secondary structure prediction. One of these features was the Shannon base-pairing entropy measure [52, 69]. Given an RNA sequence, the Shannon entropy can be computed from the ensemble of predicted secondary structures, as shown in Eqs. 9-10, where $P_{i,j}$ is the probability of base-pairing between nucleotides at sequence positions $i$ and $j$, and $n$ is the length of the RNA sequence. Note that the higher the entropy, the lower the structural prediction reliability.

$$Shannon\ base\ pairing\ entropy = \frac{1}{n}\sum_{i=1}^{n} S_i \tag{9}$$

$$S_i = -\sum_{j} P_{i,j} log(P_{i,j}) \tag{10}$$

Figure 3.1 shows the folding statistics (MFE and Shannon entropy) for each ncRNA in Positive936 compared to Dishuffle936. In agreement with [52, 25], the ncRNAs in our data set were observed to have lower MFE and Shannon entropy than their di-shuffled sequences.

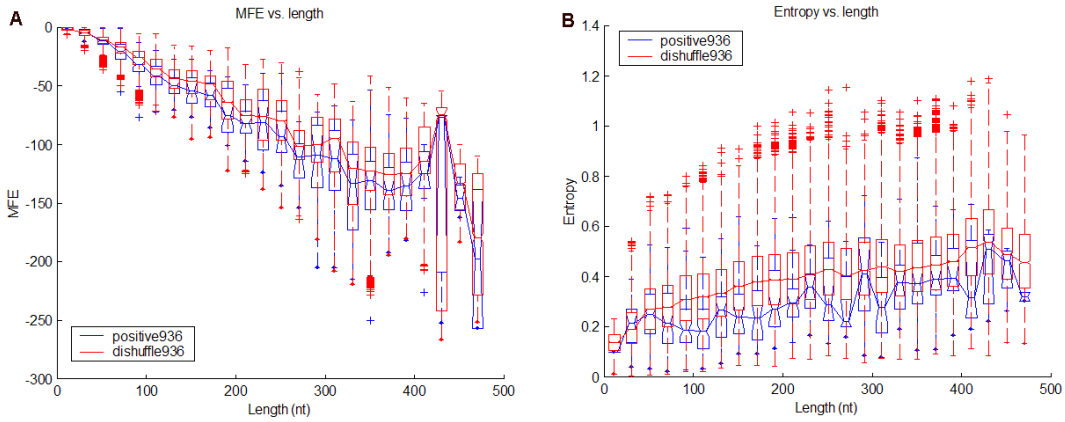


**Figure 3.1:** Boxplots for the (A) MFE and (B) Shannon entropy folding measures vs. sequence lengths for ncRNAs (Positive936) and decoys (Dishuffle936). The outliers indicated by the tick marks are values more than two times the inter-quartile range. The MFE and Shannon entropy of ncRNAs tend to be smaller than for their shuffled sequences.

#### *3.2.2.2 Ensemble statistics*

Besides the Shannon base-pairing entropy, we investigated three other ensemble-based features to assess the global folding reliability between all structures in the Boltzmann ensemble. These features included (1) the free energy of the thermodynamic ensemble, (2) the ensemble diversity statistic computed by RNAfold, and (3) the frequency of the MFE structure. These features measured the average free energy, base-pair distance, and uniqueness of the MFE structure [60]. The free energy of the ensemble for ncRNAs tended to be lower and hence more stable, while the ensemble of ncRNA structures tended to be less diverse, indicating that the structures were more unique compared to their di-shuffled sequence, as shown in Figure 3.2.

Since the prediction accuracy of secondary structures can improve substantially with the inclusion of suboptimal structures near the MFE [72], we applied an RNA secondary structure clustering algorithm, RNACluster [95], to cluster 1,000 predicted structures sampled from all possible secondary structures according to the Boltzmann equilibrium probability distribution [35]. Using the base-pairing distance between predicted secondary structures [95], we calculated various statistics to assess the cluster quality of the sampled structures. One statistic measured the compactness of each cluster (or cluster density) as defined in [95]

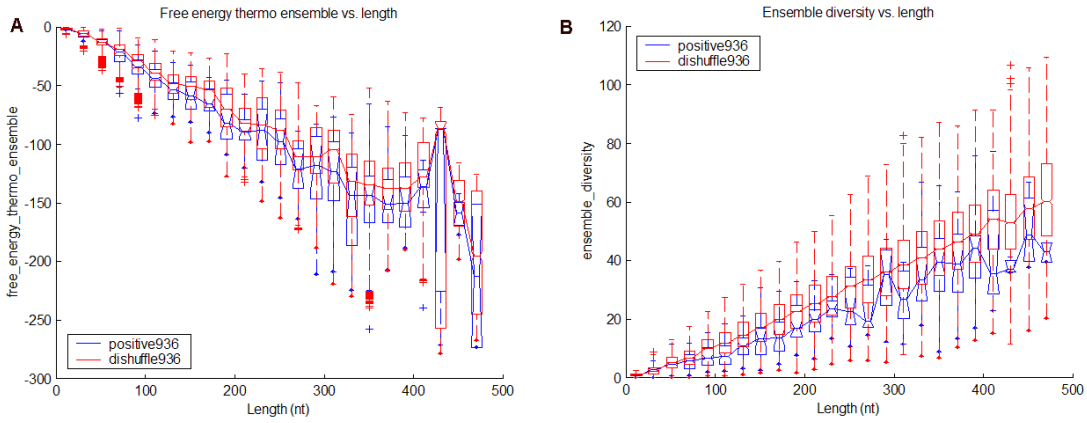


**Figure 3.2:** Ensemble statistics. Boxplots for the (A) free energy of the thermodynamic ensemble and (B) ensemble diversity folding measures vs. length for ncRNAs (Positive936) and their decoys (Dishuffle936). The outliers indicated by the tick marks are values more than two times the inter-quartile range. The free energy of the ensemble for ncRNAs tend to be lower and hence more stable. The ensemble of ncRNA structures tend to be less diverse, which indicates that their structures tend to be more unique compared to their decoys.

and shown in Eq. 11, where $d_{ij}$ is the base-pair distance and $m$ is the number of structures within a cluster.

$$compactness = \frac{\sum_i \sum_j d_{ij}}{m(m-1)} \tag{11}$$

Unlike the clustering analysis of predicted secondary structures done by the authors of Sfold [20, 33, 34], our approach used a rigorous and unique clustering method employed in RNACluster [95]. RNACluster identifies dense clusters in the space of all predicted structures by representing the structures as a minimum spanning tree (MST) and by identifying subtrees of the MST that form statistically significant clusters. We calculated five statistics, based on Chan et al. [20], for discriminating structural RNAs from their decoys using RNACluster: (1) the number of high-frequency base-pairs in the ensemble, (2) the average number of high-frequency base-pairs per cluster, (3) the average base-pair distance between the MFE structure and the ensemble, (4) the between-cluster sum of squares (BSS), and (5) the within-cluster sum of squares (WSS). The BSS statistic measures the base-pair distance between the cluster centroid and the ensemble centroid, while the WSS statistic measures the base-pair distance between the cluster centroid with all structures within that cluster

[20].

We define the centroids in a data set using the notation from Ding et al. [33] and Liu et al. [95]. Given a set of $m$ secondary structures $I_1$, $I_2$, ..., $I_m$ with $I_k = I_{ij}^k$ for $1 \leq k \leq m$, where $I_{ij} = 1$ if base $i$ pairs with base $j$ or $I_{ij} = 0$ otherwise, the centroid of a set represents a structure $\bar{I}$ where $\bar{I}_{ij} = 1$ if $\sum_{k=1}^{m} I_{ij}^k > 0.5m$ or $\bar{I}_{ij} = 0$ otherwise. For the ensemble centroid, the set is taken over the entire ensemble secondary structures, while for the cluster centroid, the set is taken over the individual cluster. These centroids define the optimal structure with the smallest base-pair distance to other structures [33]. Besides using this definition of centroid based on the optimal structure, we also selected a non-optimal structure, $I_z$, for $1 \leq z \leq m$, from the set of $m$ structures that minimizes $\sum_{k=1}^{m} D(I_z, I_k)$, where $D(\cdot\,,\cdot\,)$ is the base-pair distance. We recalculated the BSS and WSS statistics based on this non-optimal "centroid" structure, which we denoted as BSS_point and WSS_point, respectively. In addition, we incorporated the following novel statistics related to the compactness of a cluster: (i) the average compactness, (ii) the maximum compactness, (iii) the minimum compactness, (iv) the compactness of the largest cluster, and (v) the overall compactness to assess the cluster quality generated by RNACluster. Note that the average compactness is the mean of the compactness statistic over all the clusters, while the overall compactness is taken over the entire collection of structures, i.e., the sum of all the distances normalized by the number of structures in the entire collection of structures. The average compactness gives a more localized view of the density of the clusters while the overall compactness gives a more global view of the density of all the structures. Finally, we examined the number of clusters as found by RNACluster.

The statistics calculated by RNACluster were found to be highly discriminatory for separating ncRNAs from their di-shuffled versions, as shown by the P-values in Table 3.1. Using the RNACluster method, the structures of known ncRNAs tended to form fewer clusters and be more densely clustered than their di-shuffled versions, as shown in Figure 3.9A. Additional compactness-related boxplots are shown in Figure 3.3. The statistics from the largest cluster, as shown in Figure 3.4, also reflected the same trend for lower compactness statistics in the positive set compared to the di-shuffled set. Our calculation of

the relevant statistics from Chan et al. [20], utilizing RNACluster agrees with the authors' results, as shown in Figures 3.5-3.7. The main advantage of using RNACluster is that it automatically determines the optimal number of clusters through the use of the MST clustering algorithm. This approach reduced some computational complexity in having to compute the CH index, as needed in Sfold, to determine the optimum number of clusters from the hierarchical clustering method [33, 35]. We showed that our calculation of the WSS_point statistic using RNACluster was more discriminative than the WSS statistic from Sfold in Figure 3.9B and Figure 3.6B, respectively. These results are also reflected in the P-value in Table 3.1.

#### *3.2.2.3 Structural statistics*

We also considered another set of novel structural features derived from the predicted RNA secondary structures that was useful for the identification of ncRNAs. We examined various properties of known RNA secondary structural elements, i.e., stems and loops for their possible discerning power between actual ncRNAs and their di-shuffled sequences. For each stem, loop, internal-loop, and bulge structural element, as shown in Figure 3.8, we computed the 18 statistics defined in Table 3.2. These statistics included the number of structural elements present in the structure, the number of nucleotides present, and the average length of each structural element. We also examined the total internal-loop statistics, taking into account both internal loops and bulges. The number of multiloops and an estimated number of multiloop branches were also computed. It should be noted that some structural features such as stem statistics have been previously used in other applications for phylogeny studies and for identifying microRNA precursors, which have a characteristic stem-loop structure [108]. To the best of our knowledge, these features have not been applied in the *de novo* identification of ncRNAs.

From the structural statistics shown in Figure 3.10, real ncRNAs tended to have fewer stem branches, but the stems tend to be longer on average. This longer stem preference contributes to more stability in the RNA secondary structure. Real ncRNAs also tend to have more loops, as shown in Figure 3.11A. This is in agreement with the published

**Table 3.1:** P-value from Wilcoxon signed rank, rank sum, and paired t-test for all features.

| | Features | P-value (signrank) | P-value (ranksum) | P-value (ttest) |
|---|---|---|---|---|
| folding statistics | **mfe** | 4.756E-103 | 1.374E-02 | 3.997E-03 |
| | **entropy** | 9.575E-50 | 5.813E-41 | 1.221E-35 |
| | | | | |
| structural statistics | **stem_count** | 2.154E-37 | 3.268E-02 | 4.822E-02 |
| | stem_nt | 6.725E-26 | 4.814E-01 | 3.993E-01 |
| | **stem_ave** | 6.211E-43 | 4.717E-31 | 2.570E-28 |
| | **loop_count** | 1.970E-02 | 6.535E-01 | 2.274E-01 |
| | loop_nt | 1.629E-01 | 9.095E-02 | 8.799E-01 |
| | **loop_ave** | 1.061E-17 | 3.128E-29 | 1.782E-02 |
| | internal_count | 3.888E-23 | 3.601E-06 | 1.640E-05 |
| | internal_nt | 5.132E-13 | 7.364E-06 | 1.090E-03 |
| | internal_ave | 4.519E-01 | 8.555E-01 | 4.428E-01 |
| | internal_asymmetry_ave | 7.739E-09 | 5.179E-12 | 5.903E-01 |
| | bulge_count | 9.555E-13 | 1.482E-06 | 2.640E-03 |
| | bulge_nt | 9.030E-16 | 2.170E-14 | 7.385E-03 |
| | bulge_ave | 3.488E-12 | 2.125E-12 | 9.912E-01 |
| | **totalinternal_count** | 5.638E-29 | 4.998E-06 | 2.922E-05 |
| | **totalinternal_nt** | 6.404E-14 | 8.642E-05 | 8.794E-04 |
| | totalinternal_ave | 4.715E-01 | 3.273E-01 | 2.034E-01 |
| | multiloop_count | 6.600E-01 | 1.664E-06 | 3.994E-01 |
| | **multiloop_ave** | 2.258E-06 | 1.295E-02 | 1.334E-02 |
| | | | | |
| ensemble statistics | **free_energy_thermo_ensemble** | 1.433E-104 | 2.692E-02 | 9.564E-03 |
| | freq_mfe_structure_ensemble | 2.466E-03 | 2.048E-02 | 1.765E-01 |
| | **ensemble_diversity** | 1.172E-46 | 4.308E-10 | 6.646E-07 |
| | | | | |
| RNAcluster statistics | **nclusters** | 1.351E-15 | 1.349E-06 | 5.789E-07 |
| | **avecompactness** | 9.255E-61 | 4.293E-09 | 3.793E-05 |
| | **maxcompactness** | 2.084E-55 | 1.063E-09 | 1.004E-07 |
| | **mincompactness** | 3.102E-30 | 1.280E-18 | 1.710E-03 |
| | **nlargest** | 3.930E-12 | 6.785E-62 | 3.173E-03 |
| | **compactnesslargest** | 7.998E-36 | 1.415E-11 | 4.852E-07 |
| | **overallcompactness** | 1.326E-45 | 5.509E-10 | 8.303E-07 |
| | **num_hifreq_bp_ensemble** | 1.432E-36 | 1.381E-39 | 2.721E-19 |
| | **ave_num_hifreq_bp_percluster** | 2.364E-40 | 7.686E-15 | 2.179E-08 |
| | **ave_bpdist_mfe_ensemble** | 5.701E-34 | 1.001E-38 | 3.247E-26 |
| | **bss** | 2.048E-28 | 4.314E-43 | 4.318E-18 |
| | **wss** | 4.685E-18 | 4.465E-19 | 3.898E-08 |
| | **bss_point** | 4.248E-45 | 4.327E-63 | 5.429E-106 |
| | **wss_point** | 4.060E-112 | 7.352E-122 | 3.405E-73 |

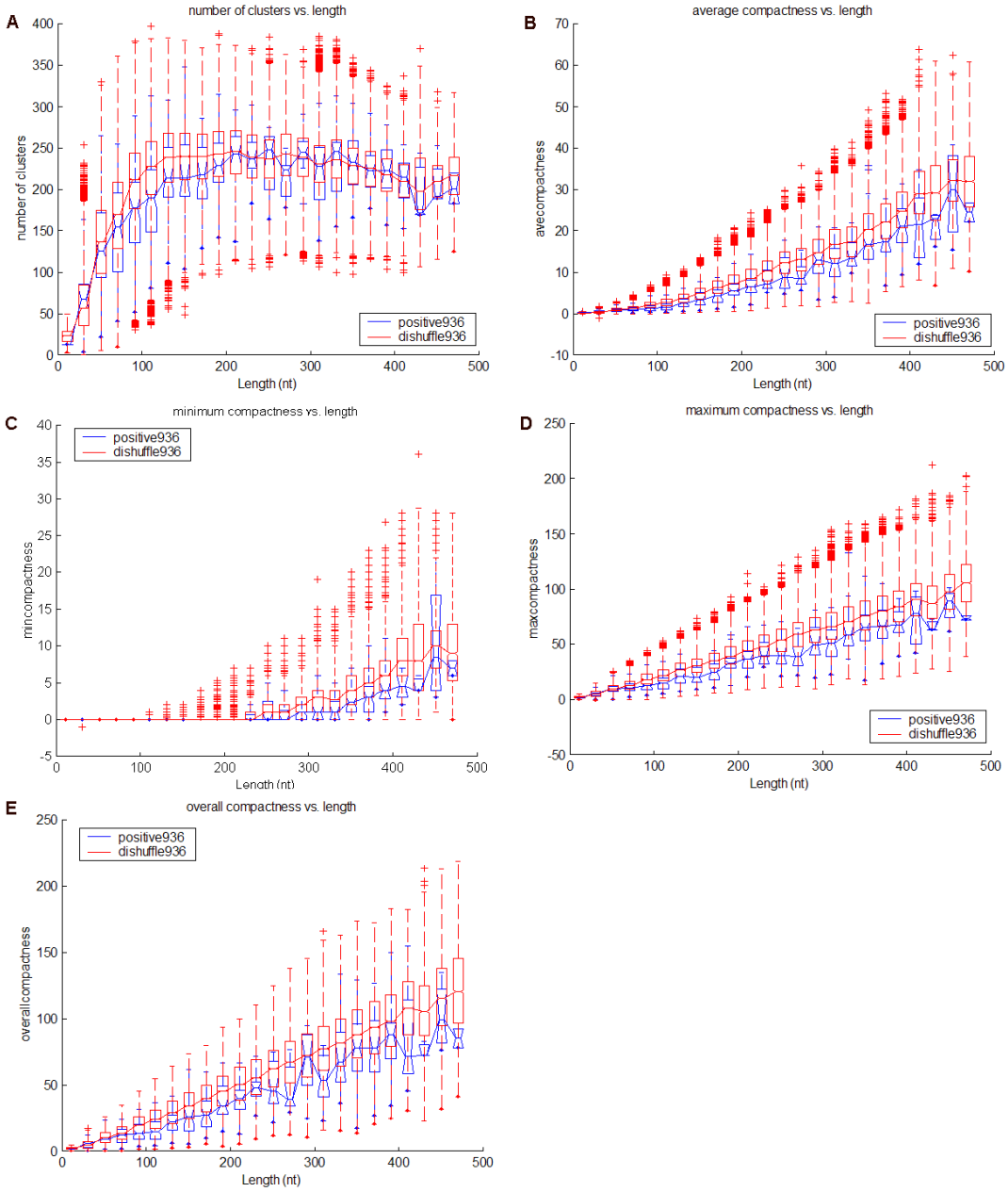


**Figure 3.3:** Ensemble statistics (RNACluster). Boxplots for the (A) number of clusters, (B) average compactness, (C) minimum compactness, (D) maximum compactness, and (E) overall compactness vs. length for ncRNAs (Positive936) and their decoys (Dishuffle936). The outliers indicated by the tick marks are values more than two times the inter-quartile range. In general, ncRNAs tend to have fewer clusters that are more dense (lower compactness measure) than their decoys.

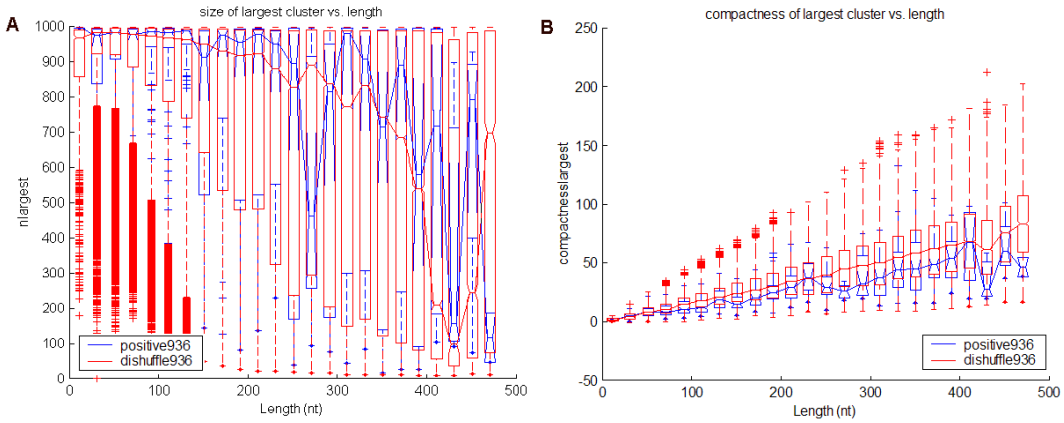


**Figure 3.4:** Ensemble statistics (RNACluster). Boxplots for the (A) size of the largest cluster and (B) compactness measure of the largest cluster vs. length for ncRNAs (Positive936) and their decoys (Dishuffle936). The outliers indicated by the tick marks are values more than two times the inter-quartile range. For most samples, the size of the largest cluster in known ncRNAs tends to be greater than their decoys. The compactness measure for the largest cluster is also consistent with Figure 3.3.

**Table 3.2:** Defined RNA secondary structural statistics and computed statistics for the RNA secondary structure from Figure 3.8.

| | | |
|---|---|---|
| stem count | # of stem branches | 5 |
| stem nt | # of nucleotides in stem base pairing | 28 |
| stem ave | average length (nt) per stem = $\frac{stem\_nt}{2 \times stem\_count}$ | 2.8 |
| loop count | # of hairpin loops | 2 |
| loop nt | # of nucleotides in hairpin loops | 9 |
| loop ave | average length (nt) per hairpin loop = $\frac{loop\_nt}{loop\_count}$ | 4.5 |
| internal count | # of internal loops (excludes bulges) | 1 |
| internal nt | # of nucleotides in internal loops (excludes bulges) | 2 |
| internal ave | average length (nt) per internal loop = $\frac{internal\_nt}{(2 \times internal\_count)}$ | 1 |
| internal asymmetry ave | average difference in size for each side of internal loop | 0 |
| bulge count | # of bulges | 1 |
| bulge nt | # of nucleotides in bulges | 1 |
| bulge ave | average length (nt) per bulge = $\frac{bulge\_nt}{bulge\_count}$ | 1 |
| total internal count | # of all internal loops = $(2 \times internal\_count) + bulge\_count$ | 3 |
| total internal nt | # of nucleotides in all internal loops = internal_nt + bulge_nt | 3 |
| total internal ave | average length (nt) per all internal loops = $\frac{total\_internal\_nt}{total\_internal\_count}$ | 1 |
| multiloop count | # multiloops | 1 |
| multiloop ave | loop_count in multiloop (estimate for # of branches) | 2 |

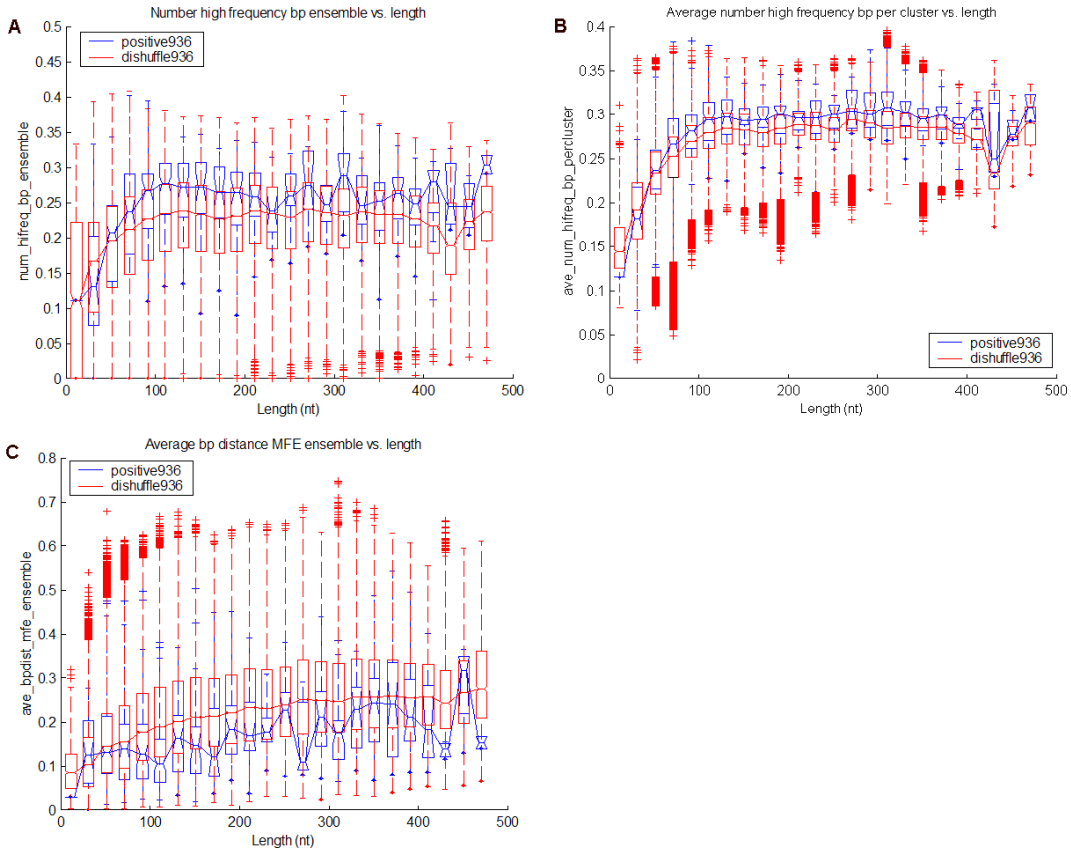


**Figure 3.5:** Ensemble statistics from Sfold computed using RNACluster. Boxplots for the (A) number of high frequency base-pairs in ensemble, (B) average number of high frequency base-pairs per cluster, and (C) average base-pair distance of MFE in ensemble vs. length for ncRNAs (Positive936) and their decoys (Dishuffle936). The outliers indicated by the tick marks are values more than two times the inter-quartile range. The results are consistent with the results from [20].

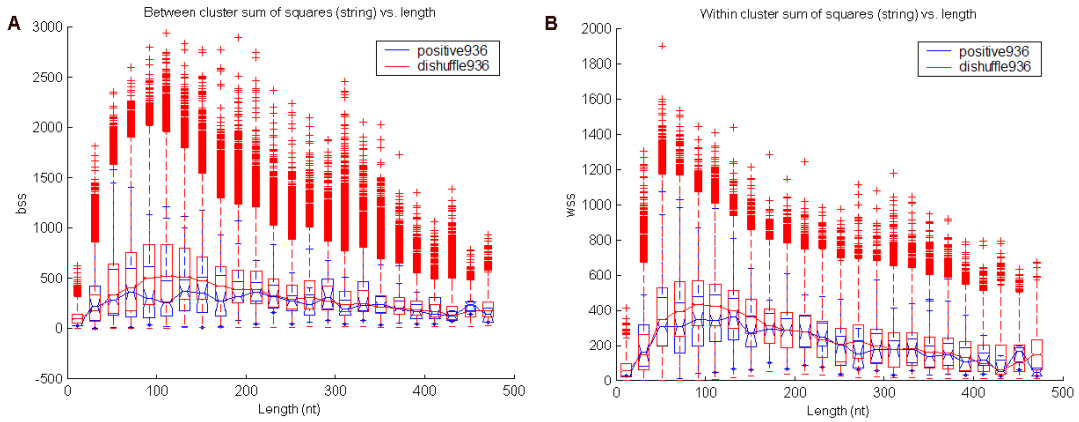


**Figure 3.6:** Ensemble statistics from Sfold computed using RNACluster. Boxplots for the (A) between-cluster sum of squares (BSS) and (B) within-cluster sum of squares (WSS) vs. length for ncRNAs (Positive936) and their decoys (Dishuffle936). The outliers indicated by the tick marks are values more than two times the inter-quartile range. The results are consistent with the results from [20].

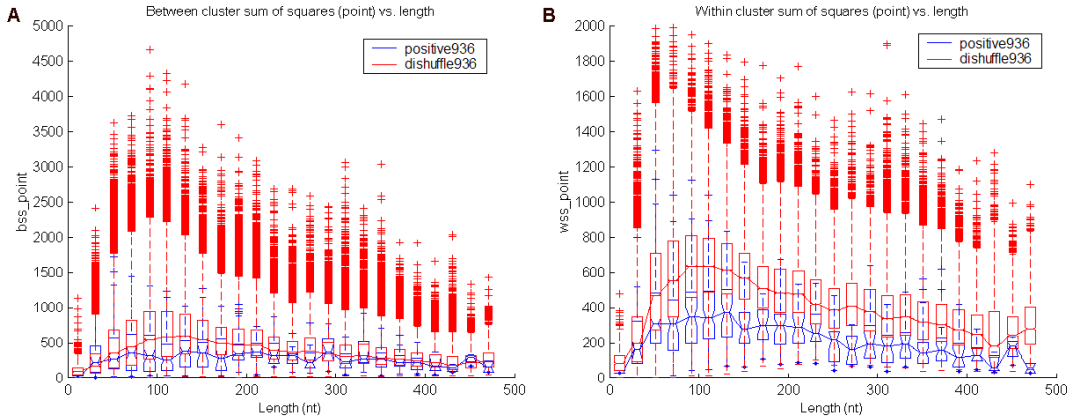


**Figure 3.7:** Ensemble statistics (RNACluster). Boxplots for the (A) between-cluster sum of squares (BSS_point) and (B) within-cluster sum of squares (WSS_point) vs. length for ncRNAs (Positive936) and their decoys (Dishuffle936). The outliers indicated by the tick marks are values more than two times the inter-quartile range. The BSS_point and WSS_point are generally smaller in known ncRNAs than in their decoys. The results are consistent with the results in Figure 3.6 and may be more discriminative on visual comparison of Figures 3.7 (B) and 3.6 (B).

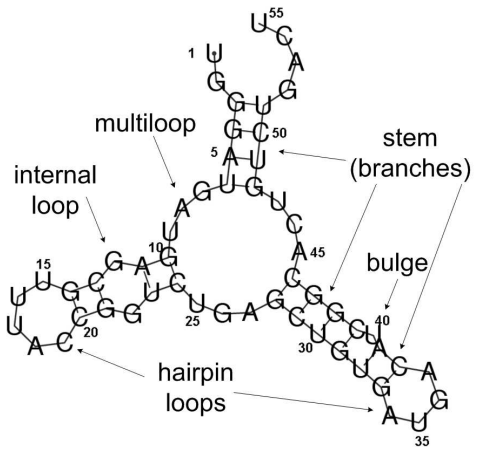


**Figure 3.8:** Basic RNA secondary structural elements consist of stems, hairpin-loops, internal-loops, and multiloops. RNA can fold onto itself by forming three, two, or one hydrogen bond(s) between nucleotide pairs C-G, A-U, and G-U pairs, respectively. The stems are regions with hydrogen bond base pairing represented by a connecting line. Consecutive stem base pairings are called branches. Single-stranded regions with no base pairing belong to hairpin-loops, internal-loops, or multiloops. Hairpin-loop regions are single-stranded segments of the secondary structure closed by exactly one branch, whereas internal-loops are closed by exactly two branches. Multiloops are special cases in which three or more branches are connected. A bulge is a special case of an internal-loop in which one side does not have extra single-stranded bases.

literature where the single-stranded regions of loops can play an important role in RNA-protein interaction [2, 63, 75, 38]. In addition, loops often times contain regions of target complemetarity between ncRNAs and their mRNA targets in microbes. The presence of more loops may also be related to the functional role of the ncRNAs. When multiloops are present, there tended to be more loops in real ncRNAs than in their di-shuffled version, as shown in Figure 3.12. Not all single-stranded regions were more dominant in real ncRNAs. As seen in Figure 3.11B, the total internal-loops consisting of internal loops and bulge regions were actually less in ncRNAs than in their di-shuffled sequences. This tendency for ncRNAs to have fewer of such structural elements may have some functional interpretation that can be applied to ncRNA gene finding. Additional boxplots for loop-related structures are shown in Figures 3.13-3.14.

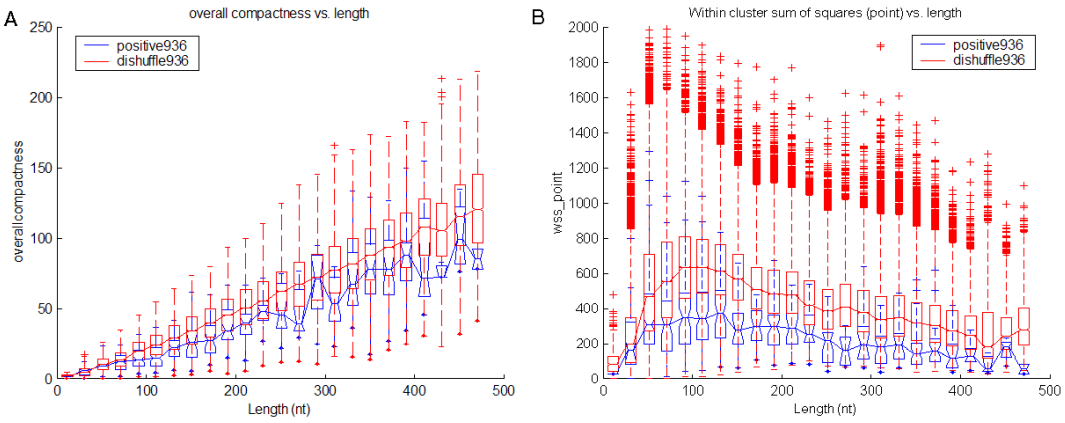


**Figure 3.9:** Ensemble statistics. Boxplots for the (A) overall compactness and (B) within cluster sum of squares vs. sequence lengths for ncRNAs (Positive936) and their decoys (Dishuffle936). The outliers indicated by the tick marks are values more than two times the inter-quartile range. In general, ncRNAs tend to have fewer clusters that are denser (lower compactness measure) than their decoys and their within-cluster sum of squares is generally smaller than that of their decoys.

#### *3.2.2.4 Significant features*

For all the features examined above, we used hypothesis testing to identify those features that can potentially distinguish known ncRNAs from their di-shuffled sequences. We performed a paired t-test, comparing the mean of the features from the Positive936 data set with the mean from the Dishuffle936 data set, and computed the P-value estimating the probability that these samples have the same means, as summarized in Table 3.1. Since the t-test assumes distributions of equal variances, we also computed the significance according to the Wilcoxon signed rank and rank sum test not based on this assumption and found similar results. We manually selected a set of 25 features with significant P-values below 0.05, which we refer as the f25 feature set. This set included two folding statistics, two ensemble statistics, 14 RNACluster statistics, and seven structural statistics, as shown in Table 3.1. All features were length normalized (when applicable) before using them for genome-wide prediction.

### 3.2.3 Application to genome-wide prediction

We extracted all 93 known ncRNAs for *E. coli* from the Positive1540 data set. Using these ncRNAs as queries, we ran an all-versus-all BLASTN search against the Positive936

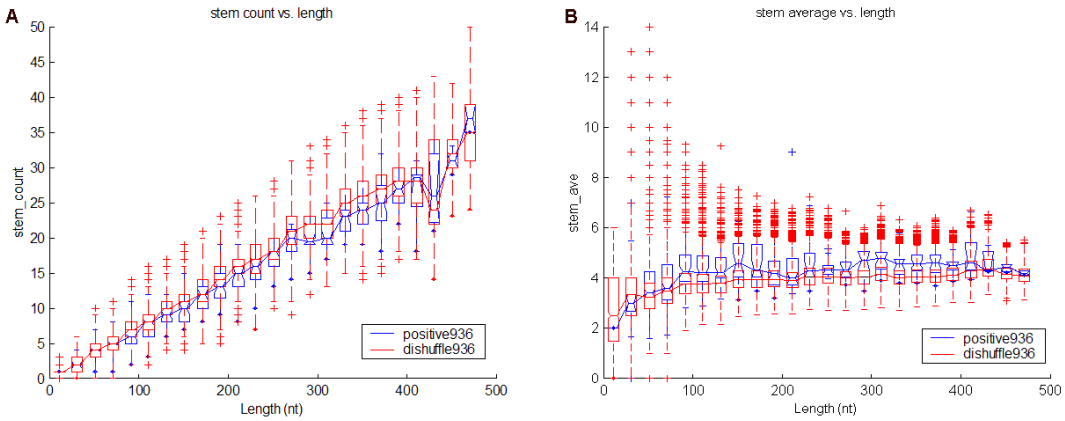


**Figure 3.10:** Structural statistics. Boxplots for the (A) stem count and (B) stem average vs. lengths for ncRNAs (Positive936) and their decoys (Dishuffle936). The outliers indicated by the tick marks are values more than two times the inter-quartile range. In general, ncRNAs tend to have fewer stem regions while each stem region is longer on average than their decoys.

data set and removed all the Positive936 hits below an E-value cutoff of $10^{-5}$. We reduced the original Positive936 data set to 800 unique ncRNAs after removing sequences homologous to the 93 known ncRNAs. We then used this data set without known ncRNAs in *E. coli* for training and refer to it as Positive800_ecoli. We explored four different sets as the negative training data sets for training our classifiers. The prediction consistency among the trained classifiers based on these different negative sets helped to enhance confidence in our feature-based prediction of ncRNAs. Negative_Set1 used the di-shuffled sequences of Positive800_ecoli. Negative_Set2 consisted of di-shuffled randomly selected sequence segments in *E. coli* to ensure that no ncRNA-related secondary structures were present. Negative_Set3 took di-shuffled random segments in *E. coli* corresponding to where samples of Positive800_ecoli were found, i.e., intergenic, protein-overlapping, or antisense to protein-coding region. Negative_Set4 was similar to Negative_Set3 but the sequences were not di-shuffled. Additional details on each negative set are shown in Table 3.3.

For each negative training set, we computed all the f25 significant features and an additional 20 sequence-based statistics, namely, four mono- and 16 di-mer frequencies because they were useful in distinguishing between real ncRNAs and decoys by previous algorithms [81, 132, 18, 153, 94]. We used a feature-ranking procedure based on the F1 score to assess

**Table 3.3:** Negative sets tested using *E. coli* genome.

| |
|---|
| Negative_Set1 (Dishuffle ncRNAs)<br>If $s_i$ represents the $i^{th}$ sequence where $1 \leq i \leq 800$, Negative_Set1 was generated by dishuffling each sample in Positive800_ecoli, i.e., dishuffle($s_i$). This shuffling procedure allowed the mono- and di-nucleotide frequencies to be preserved while disrupting any secondary structure present in the positive set. |
| Negative_Set2 (Dishuffle random positions in organism)<br>For Negative_Set2, we generated a negative sample corresponding to dishuffle(random(genome(Ecoli),length($s_i$))), which randomly selects a genomic segment in the *E. coli* genome of length corresponding to the original Positive800_ecoli and then apply the dishuffling strategy. This negative set approach allows for base composition differences among various organisms to be taken into account but at the same time distorting any sort of secondary structure that may be present in the random sampling. |
| Negative_Set3(Dishuffle random positions in organism sampled with *a priori* genomic location)<br>For each sample in the Positive800_ecoli data set, Negative_Set3 was generated as a function of dishuffle(random(genome(Ecoli),length($s_i$),caseid($s_i$))) where the random sample in *E. coli* is taken as a function of the length of the original sequence and where caseid($s_i$) can be 'intergenic', 'CDS_same', 'antisense', or 'other'. 'Intergenic' corresponds to those regions in which the ncRNA does not overlap any annotated protein-coding genes. 'CDS_same' refers to the case when the ncRNA overlaps an annotated protein-coding gene transcribed on the same strand. 'Antisense' is the case when the ncRNA overlaps an annotated protein-coding gene transcribed in the opposite strand. The 'other' case captures any other case not previously caught, e.g., when an ncRNA could overlap protein-coding genes on both the same strand and the opposite strand. This negative set allows for the sampling to be more constrained to specific regions of the genome by imposing the additional condition of caseid. |
| Negative_Set4(Random positions in organism sampled with a priori genomic location)<br>Negative_Set4 is similar to Negative_Set3; however, we do not perform the dishuffling strategy but instead use the sampled sequence from the organism, i.e., the negative set is defined as random(genome(Ecoli),length($s_i$),caseid($s_i$)). This negative set was generated so that we can assess the performance of the shuffling procedure preserving mono- and di-nucleotide frequencies. |

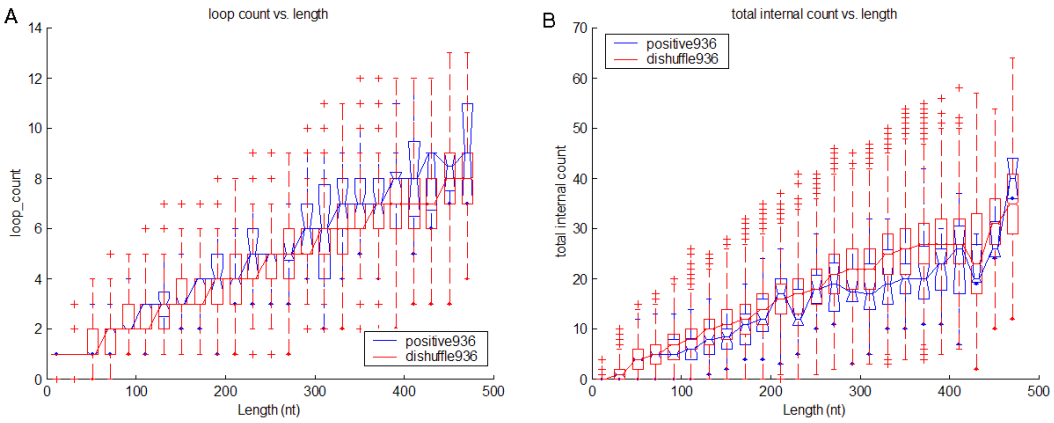


**Figure 3.11:** Structural statistics. Boxplots for the (A) hairpin-loop count and (B) total internal-structure count (internal-loop and bulges) vs. lengths for ncRNAs (Positive936) and their decoys (Dishuffle936). The outliers indicated by the tick marks are values more than two times the inter-quartile range. In general, ncRNAs tend to have more loop regions and fewer internal-loops on average than their decoys.

the discrimination power as defined in Eq. 12 [108], where $m$ and $s$ represent the mean and standard deviation of the positive (p) and the negative (n) distribution, respectively. Higher F1 scores indicated features with higher discriminative power between the positive and the negative sets. The top t features, as ranked by the F1 score, were used to train a neural network-based classifier to discriminate Positive800_ecoli from a given negative set.

$$F1\_score = |\frac{\mu_p - \mu_n}{\sigma_p + \sigma_n}| \tag{12}$$

Neural networks (NN) are a class of machine learning algorithms, widely used for solving classification problems based on multiple sources of information without assuming the underlying relationships among the individual information sources. This technique is robust for noisy data and has been widely used for many biological data analysis problems [18, 167, 145].

We trained our NN-based classifier using MATLAB's NN toolbox using the input features derived from our data set. The network parameters were optimized using the Levenberg-Marquardt algorithm to obtain the desired binary (1/0) classification label depending on whether each sample contains an ncRNA or not. Our classifier has a single layer, one-neuron architecture using a logsig activation function. Other NN architectures with more neurons

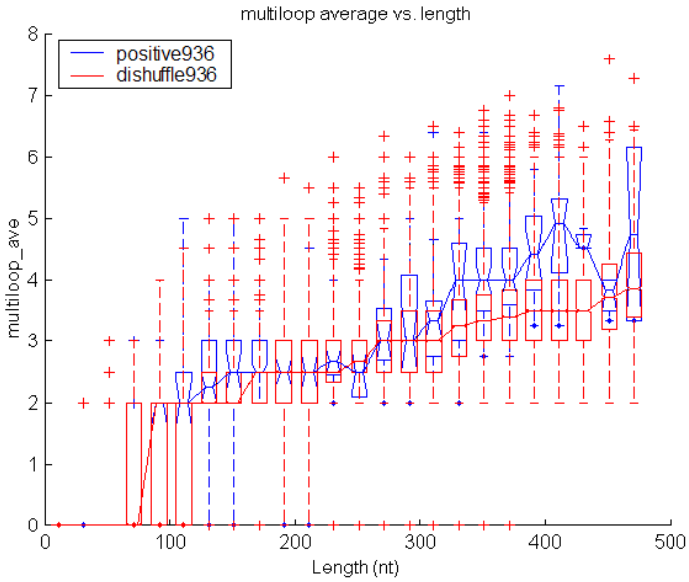


**Figure 3.12:** Structural statistics. Boxplots for the multiloop average vs. length for ncRNAs (Positive936) and their decoys (Dishuffle936). The outliers indicated by the tick marks are values more than two times the inter-quartile range. In general, ncRNAs tend to have higher number of multiloops (branches) than their decoys.

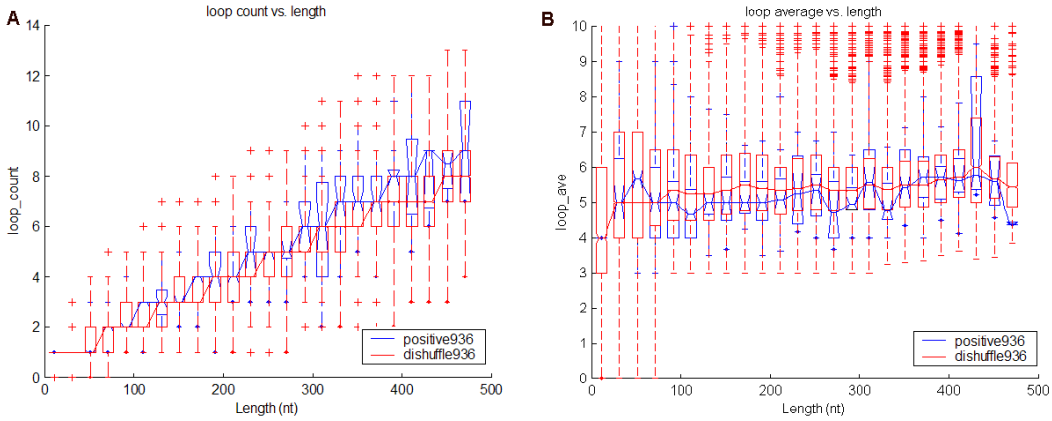


**Figure 3.13:** Structural statistics. Boxplots for the (A) loop count and (B) loop average vs. length for ncRNAs (Positive936) and their decoys (dishuffle936). The outliers indicated by the tick marks are values more than two times the inter-quartile range. In general, ncRNAs tend to have more number of loop regions and each loop region is shorter on average than their decoys.

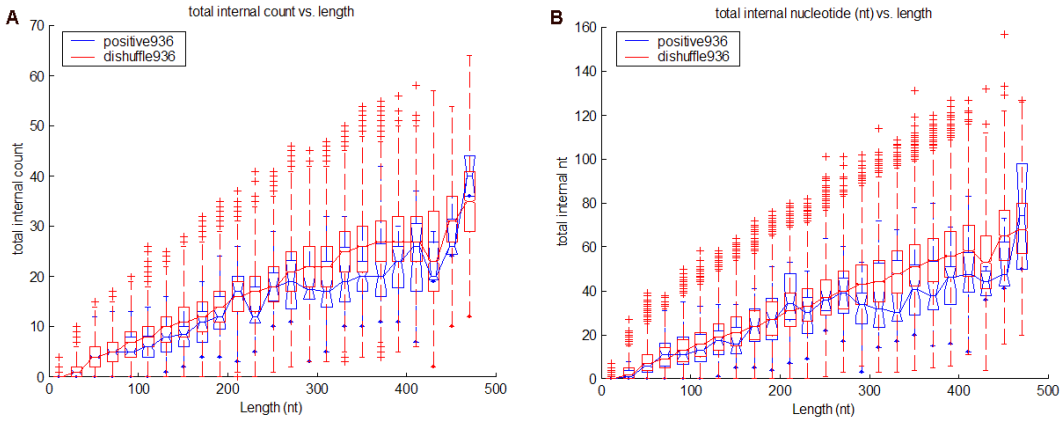


**Figure 3.14:** Structural statistics. Boxplots for the (A) total internal count and (B) total internal nucleotide vs. length for ncRNAs (Positive936) and their decoys (Dishuffle936). The outliers indicated by the tick marks are values more than two times the inter-quartile range. In general, ncRNAs tend to have fewer internal-loops and each internal-loop is shorter in length than their decoys.

in the current one-layer and two- and three- layer networks were also examined, but the performance improvements were negligible (data not shown). We tested each negative set by applying its corresponding trained NN to genome-wide prediction of ncRNAs in *E. coli K12* (NC_000913). To do this, we divided the whole genome into sequences of overlapping windows with lengths $w = 40, 80, \ldots,$ and 280 nt, on both the forward and the reverse strands, where consecutive windows overlap by $w/2$, as illustrated in Figure 3.16. These window lengths were reflective of the typical lengths of known ncRNAs, whose distribution in *E. coli* is shown in Figure 3.15. For each window, the features were computed and a class label was assigned depending on whether or not the window overlapped an ncRNA. A schematic of the training/testing procedure is summarized in Figure 3.16.

#### *3.2.3.1 Comparison of negative sets*

The training results based on the four different negative sets are shown in Table 1, where the largest area under the receiver operating curve (AUROC) is shown. The AUROC gives a measure of the prediction performance of the trained classifier, independent of a specific threshold. In general, a larger AUROC reflects higher sensitivity and higher specificity. A smaller AUROC was observed for Negative_Set1 compared to the others since the latter

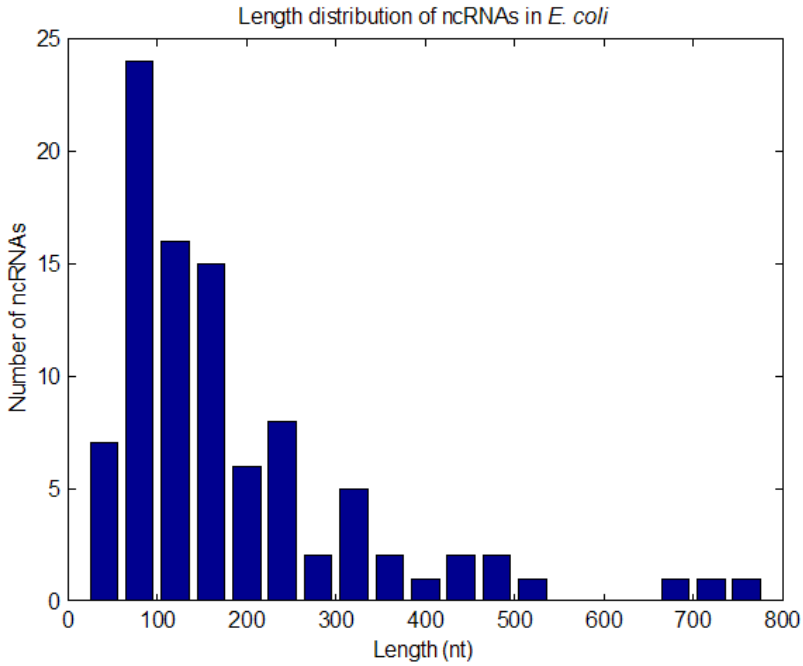


**Figure 3.15:** Histogram plot of the number of ncRNAs at each length contained in the known *E. coli* set.

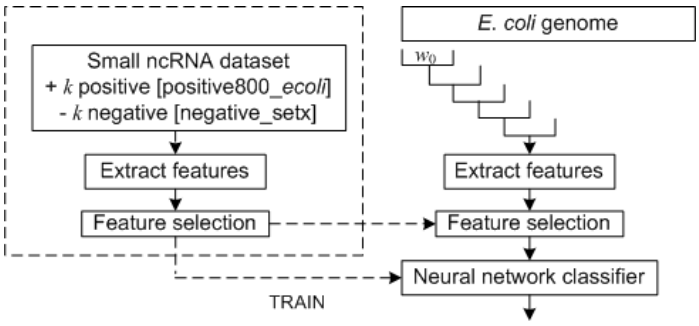


**Figure 3.16:** Schematic of the training/testing procedure for *de novo* genome-wide prediction of ncRNA genes. For each negative set, the features were extracted for each sample $i$ of the positive and negative set for $1 \leq i \leq k$. Each sliding window, $w$, and the corresponding features were used as inputs to our NN-based classifier to predict if that sequence has the potential to contain an ncRNA gene.

benefits from having training samples with organism-specific structure and sequence information. Negative_Set2 gave the best performance when using 25 features, which can be found in Table 3.4. Repeated simulations on five different instances of Negative_Set2 gave similar AUROC values with an overall mean of 0.6438 and a standard deviation of 0.0115. Also, if we used all 45 features available, the performance of our trained classifier based on Negative_Set2 and Negative_Set3 was comparable (data not shown), thus justifying our use of Negative_Set2 for additional analyses. Furthermore, Negative_Set3 added the complexity and an additional assumption on the prior distribution of where ncRNAs were found, e.g., intergenic, within protein-coding regions or antisense to protein-coding regions. Negative_Set3 resulted in a marginal improvement in AUROC over Negative_Set4, implying that the di-shuffling procedure does not bias the feature discrimination for identifying ncRNAs.

#### *3.2.3.2 Meta-learner classifier to combine information from different window sizes*

From Table 3.5, we noted that the prediction performance of our trained classifier varied depending on the window size used. We speculated that by combining the prediction performance across the different window sizes, we could further improve the prediction performance. By doing so, it enabled each classifier of a different window size to distinguish the positive (ncRNAs) from the negative training data for genes of different lengths. Hence, we modified the training/testing procedure slightly by training the classifier for each window size separately and then combining the prediction results of each classifier using a voting classifier, called a meta-learner, as summarized in Figure 3.17. We omit further technical details about dealing with overlapping windows of different sizes.

#### *3.2.3.3 Filter using conservation, promoter, terminator, and positional information*

To reduce the false positive rate in our predictions, we explored various filtering strategies based on other available data from *E. coli*. Like other approaches, we analyzed our predictions in conjunction with other sequence-level signals such as sequence conservation, promoter, and terminator information [6, 21].

For each prediction, we ran BLASTN to search against GenBank and collected all the hit sequences with E-value $< 10^{-5}$. We also used the promoter and transcription factor binding

**Table 3.4:** Features used in Negative_Set2 as sorted by the F1 score, the higher the score the more discriminative the feature.

| # | Feature | F1 |
|---|---|---|
| 1 | kmer_GA | 0.4854 |
| 2 | kmer_AG | 0.4089 |
| 3 | kmer_CA | 0.4067 |
| 4 | kmer_GG | 0.3728 |
| 5 | kmer_G | 0.3638 |
| 6 | structuralstatistics_mfe | 0.3022 |
| 7 | diversity_free_energy_thermo_ensemble | 0.2941 |
| 8 | kmer_UU | 0.2264 |
| 9 | rnacluster_ave_num_hifreq_bp_percluster | 0.2261 |
| 10 | entropy_entropy | 0.2257 |
| 11 | rnacluster_num_hifreq_bp_ensemble | 0.2115 |
| 12 | kmer_U | 0.1970 |
| 13 | rnacluster_overallcompactness | 0.1945 |
| 14 | rnacluster_avecompactness | 0.1909 |
| 15 | structuralstatistics_stem_ave | 0.1858 |
| 16 | diversity_ensemble_diversity | 0.1807 |
| 17 | kmer_GC | 0.1761 |
| 18 | kmer_C | 0.1742 |
| 19 | kmer_UC | 0.1741 |
| 20 | rnacluster_maxcompactness | 0.1677 |
| 21 | kmer_CG | 0.1622 |
| 22 | rnacluster_ave_bpdist_mfe_ensemble | 0.1591 |
| 23 | kmer_AC | 0.1462 |
| 24 | kmer_AU | 0.1190 |
| 25 | rnacluster_mincompactness | 0.1084 |
| 26 | kmer_CU | 0.1061 |
| 27 | rnacluster_compactnesslargest | 0.1042 |
| 28 | rnacluster_nlargest | 0.0957 |
| 29 | kmer_CC | 0.0934 |
| 30 | structuralstatistics_loop_ave | 0.0918 |
| 31 | rnacluster_bss_point | 0.0902 |
| 32 | kmer_UG | 0.0891 |
| 33 | rnacluster_bss | 0.0881 |
| 34 | kmer_AA | 0.0732 |
| 35 | kmer_GU | 0.0711 |
| 36 | kmer_UA | 0.0677 |
| 37 | rnacluster_nclusters | 0.0623 |
| 38 | rnacluster_wss_point | 0.0331 |
| 39 | rnacluster_wss | 0.0285 |
| 40 | structuralstatistics_multiloop_ave | 0.0143 |
| 41 | structuralstatistics_stem_count | 0.0139 |
| 42 | structuralstatistics_total_internal_nt | 0.0138 |
| 43 | kmer_A | 0.0062 |
| 44 | structuralstatistics_total_internal_count | 0.0060 |
| 45 | structuralstatistics_loop_count | 0.0015 |

**Table 3.5:** AUROC values for the four negative training sets. Each row represents the AUROC values for a different window size, $w$, using both the forward and reverse strands.

| | Negative_Set1 | Negative_Set2 | Negative_Set3 | Negative_Set4 |
|---|---|---|---|---|
| | f25 | f25 (F1) | f45 | f45 |
| 40 | 0.5275 | 0.6135 | 0.5801 | 0.5644 |
| 40_rc | 0.5185 | 0.6163 | 0.5967 | 0.6112 |
| 80 | 0.5571 | 0.6485 | 0.5327 | 0.6020 |
| 80_rc | 0.5141 | 0.6389 | 0.6258 | 0.6087 |
| 120 | 0.5680 | 0.6714 | 0.6548 | 0.6154 |
| 120_rc | 0.5123 | 0.6258 | 0.5925 | 0.5728 |
| 160 | 0.5683 | 0.6636 | 0.5000 | 0.6269 |
| 160_rc | 0.5151 | 0.6332 | 0.6254 | 0.5798 |
| 200 | 0.5530 | 0.6681 | 0.6324 | 0.6047 |
| 200_rc | 0.5238 | 0.6473 | 0.6074 | 0.5712 |
| 240 | 0.5858 | 0.6977 | 0.5544 | 0.5466 |
| 240_rc | 0.5244 | 0.6498 | 0.6452 | 0.5845 |
| 280 | 0.5720 | 0.6827 | 0.6666 | 0.6001 |
| 280_rc | 0.5272 | 0.6538 | 0.6407 | 0.5539 |
| mean | 0.5405 | 0.6508 | 0.6039 | 0.5887 |

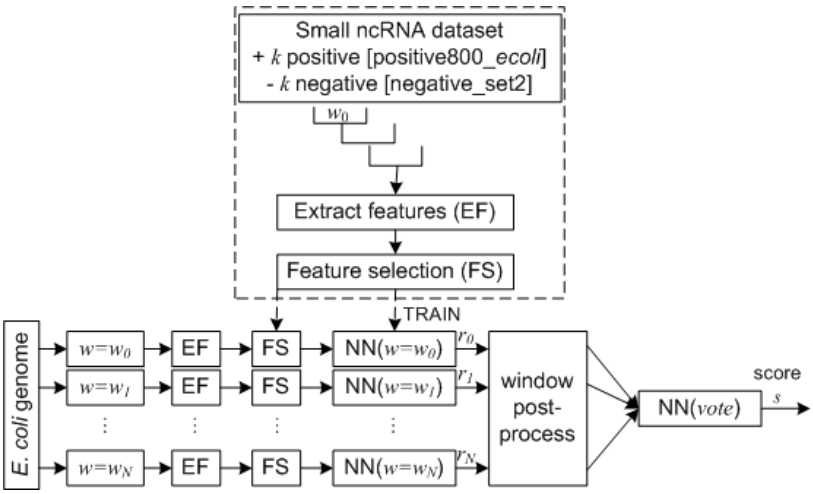


**Figure 3.17:** Schematic of classifier architecture used for genome-wide prediction. The results of each NN-based classifier are then post-processed and combined into a final NN-based classifier to make the final prediction. The output of the length-specific NN-based classifiers and voting classifier are labeled by score $r_i$ for $0 \leq i \leq N$ and score $s$, respectively.

site information from RegulonDB [130] to compile promoter regions within 300 nucleotides upstream of the predicted ncRNA. TransTermHP [79] was used to predict rho-independent transcription terminators downstream of our predicted ncRNAs.

In addition, we analyzed the position of known ncRNAs in *E. coli* by classifying each one into four classes: (1) intergenic, (2) cds_samestrand, (3) antisense, and (4) other cases. Antisense and cds_samestrand cases corresponded to occurrences when an ncRNA overlapped with an annotated protein-coding region on the antisense strand or the same strand, respectively. The other cases capture special situations where sections of an ncRNA overlapped both an antisense and a cds_samestrand case.

Our analysis of the known ncRNA genes in *E. coli* indicated that 52% were found in intergenic regions. This location bias could be due to the fact that previous studies on ncRNA genes have been mostly focused on intergenic regions. Antisense, cds_samestrand, and other cases represent 26%, 19%, and 3% of the known ncRNA genes in *E. coli*, respectively. Antisense and cds_samestrand cases were further subcategorized into those ncRNAs that fully or partially overlapped a protein-coding region. We focused on the partially overlapping subcategories because (1) in both Positive800_ecoli and our *E. coli* data set, the partially overlapping case was approximately twice as common as the fully overlapping case and (2) experimental validation of fully overlapping cases is difficult [68]. For the partially overlapping cases, we computed the log likelihood score using Eq. 13, where ntoverlap is the number of nt in overlap between the ncRNA and the protein-coding region. The log likelihood for the antisense and cds_samestrand cases with partial overlap is shown for the Positive800_ecoli dataset in Figure 3.18. We noted that ncRNA genes partially overlapped protein-coding regions by no more than ∼50 nt, which is good for discriminating between the positive and negative sets.

We exhaustively tested different combinations of conservation, promoter, terminator, and positional cases based on the above observations to obtain our final list of candidate ncRNA genes.

$$LL(nt_{overlap}(ncRNA, CDS)) = ln \frac{P(nt_{overlap}(ncRNA, CDS)|TP)}{P(nt_{overlap}(ncRNA, CDS)|TN)} \tag{13}$$

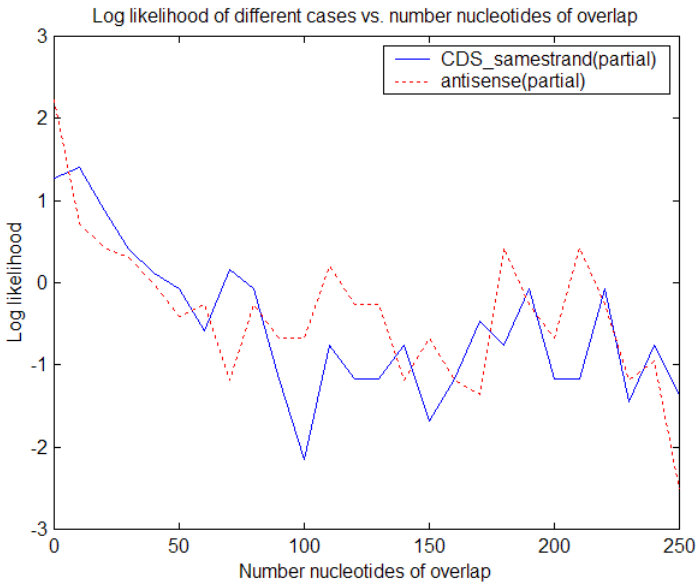


**Figure 3.18:** Log likelihood for the cds_samestrand and antisense cases with partial overlap for the Positive800_ecoli data set. A positive log likelihood score for nucleotide overlap $<$ 50 nt indicates that ncRNAs tend to overlap protein-coding genes by less than this cutoff.

#### *3.2.3.4 Filtering with tiling array data to identify candidates for experimental validation*

To identify a manageable list of candidates for experimental validation, we employed data from a high-density tiling array. Such an array permits an unbiased analysis of complete genomic transcription, including ncRNAs. By comparing our predictions to tiling array candidates, we significantly reduced the number of potentially false positive predictions. The whole genome-tiling array data set was derived by comparing an RNase E deletion strain of *E. coli* with a wild type control (Stead, M., Marshburn, S., Castillo, L.P., Ray, D.,van Bakel, H., Hughes, T., and Kushner, S., manuscript in preparation). This strain was chosen because RNase E has been shown to play an important role in general RNA metabolism in *E. coli* [11, 113, 11]. The authors identified 402 possible ncRNA candidates based on increased steady-state RNA levels in the RNase E deletion strain compared to a wild type control (Stead, M., Marshburn, S., Castillo, L.P., Ray, D.,van Bakel, H., Hughes, T., and Kushner, S., manuscript in preparation). Overall, we filtered our program's predictions based on the following conditions: (1) the potential ncRNA was conserved; (2) it contained either a predicted promoter or terminator; (3) its overlap with a protein-coding region (if

applicable) was < 50 nt; and, (4) it overlapped candidates derived from the tiling array.

#### *3.2.3.5 Bacterial strains, isolation of total RNA and Northern analysis*

The *E. coli* strains used in this study were MG1693 (*thyA715 rph-1*), which was provided by the *E. coli* Genetic Stock Center (Yale University) and an isogenic derivative, SK3564 *(rne*$\Delta$*1018::bla thyA715 rph-1 recA56 srlD::Tn10/pDHK30(rng-219* $Sm^r/Sp^r$*)/pWSK129* ($Km^r$) which has been described previously [106]. Both strains were grown in Luria broth supplemented with thymine (50 $\mu$g/ml) at 37°C. For MG1693, cells were harvested at 3.5, 6, 8, and 10 hours post-inoculation, corresponding to mid-log, early stationary, mid stationary and late stationary phase growth. For SK3564 the cells were grown in the same manner, but in order to account for its slower growth rate, were harvested at 11.5, 17.5, 20 and 23 hours post-inoculation. Harvested cells were mixed with an equal volume of crushed frozen TM buffer (10 mM Tris [pH 7.2]/5 mM $MgCl_2$) containing 20 mM $NaN_3$ and 0.4 mg/ml chloramphenicol [109]. The cells were then centrifuged at 5,000 rpm for 10 min at 4°C. The cell pellets were subsequently resuspended in Trizol (Invitrogen) and total RNA was extracted according to the manufacturer's instructions. The RNA samples were treated with DNase I using a DNA-free kit$^{TM}$ (Ambion), ethanol precipitated, quantitated with a Nanodrop apparatus (NanoDrop Technologies) and visualed on 1.0% agarose gels. For Northern analysis, 30 $\mu$g of total RNA was loaded in each lane and separated on either 6% or 8% polyacrylamide/8.3 M urea gels [150] and subsequently transferred onto Magnacharge nylon membranes (GE Water & Processing technologies) by electroblotting (1 h, 80 V, 4°C). Membranes were prehybridized in ULTRAhyb Ultrasensitive Hybridization Buffer (Ambion) at 68°C and probed with internally labeled, in vitro transcribed RNA oligomers (oligonucleotide sequences used to generate the probes are available on request). The membranes were washed twice with 2X SSC/0.1% SDS at room temperature for five minutes each and then twice with 0.1X SSC/0.1% SDS at 68°C for 15 minutes each. Hybridization was visualized on a Storm 840 PhosphorImager (Molecular Dynamics).

By utilizing folding, ensemble, and structure-based features, we developed our NN-based meta-learner for the *de novo* search of ncRNAs on a genome-wide scale. We compared the

prediction results of our method in *E. coli* to existing programs relying on homology and other information and found that our results are as good or in some cases better than these methods.

### 3.2.4 ncRNA prediction in *E. coli*

Table 3.6 summarizes the detailed prediction performance of the meta-learner. Our trained meta-learner achieved an average prediction sensitivity of 68%, specificity of 70%, and an overall accuracy of 70% for predicting windows containing ncRNAs in *E. coli*. By combining prediction results from individual window-specific NN-based classifiers, our meta-learner improved the prediction performance of the best individual window-specific classifier by ∼10% as measured by the AUROC values. The optimal AUROC performance was achieved using three window sizes, $w$ = 100, 120, and 160 nt, corresponding to three peaks of the ncRNA-length distribution in *E. coli*. If no such prior knowledge for a target genome is available, the user could use the default seven window sizes of 40, 80, 120, 160, 200, 240, and 280 nt to combine the scores into a meta-learner. Although the performance using these seven windows yielded an average sensitivity of 73%, specificity of 52%, and overall accuracy of 52%, it was still better compared to using any single window size. The trade-off for higher sensitivity is lower specificity in the forward strand. The accuracy approximates the specificity since the size of the true negative set is substantially larger than the size of the true positive set.

We used the results from the three window sizes to further analyze the predictions for *E. coli*. The AUROC curve for the *E. coli* test performance is given in Figure 3.19. For other organisms, users can select a threshold necessary to obtain a desired sensitivity and specificity trade-off for their application. For example from Figure 3.19, if we require a high specificity >90% for our application, we can choose a more stringent threshold of 0.75 to give sensitivities of 44% and 33% on the direct and reverse strands, respectively. These performance measurements assess the ability of our classifier to identify windows with the potential to contain an ncRNA.

We then obtained a unique list of candidates for the genome by labeling continuous

**Table 3.6:** Performance of our ncRNA gene prediction using different combinations of window sizes. Rows 2 and 3 show prediction results from three window sizes, $w = 100$, 120, 160. Rows 4 and 5 show prediction results from seven window sizes, $w = 40$, 80, 120, 160, 200, 240, 280. The performance is separately given for the forward and reverse strand in terms of true positive (TP), false positive (FP), false negative (FN), and true negative (TN). The prediction sensitivity, specificity, and accuracy are computed as Sn=TP/(TP+FN), Sp=TN/(TN+FP), and (TP+TN)/(TP+FN+FP+TN), respectively. All these prediction results are based on using the same NN output threshold from training. It should be noted that the AUROC was computed independent of threshold.

| # win | strand | TP | FP | FN | TN | Sn | Sp | Acc | AUROC |
|---|---|---|---|---|---|---|---|---|---|
| 3 | plus | 395 | 77802 | 155 | 153630 | 0.7182 | 0.6638 | 0.6640 | 0.7557 |
| 3 | minus | 328 | 63072 | 180 | 168402 | 0.6457 | 0.7275 | 0.7273 | 0.7628 |
| 7 | plus | 440 | 127456 | 110 | 103976 | 0.8000 | 0.4493 | 0.4501 | 0.7014 |
| 7 | minus | 331 | 94896 | 177 | 136578 | 0.6516 | 0.5900 | 0.5902 | 0.6548 |

regions with NN scores above the user chosen threshold. For example, when we used the training threshold, we predicted a total of 16,571 ncRNAs (8,712 on the forward strand, and 7,859 on the reverse strand) for the entire *E. coli* genome. Within this set, 51 candidates on the forward strand and 42 candidates on the reverse strand overlapped a known ncRNA gene. After accounting for overlap to a known ncRNA gene, these positive candidates corresponded to 47 out of 51 ncRNAs on the direct strand and 37 out of 42 ncRNAs on the reverse strand, giving rise to a prediction sensitivity of 90% (Sn=TP/(TP+FN)=(47+37)/(51+42)) and a positive predictive value (PPV) of 0.56% (PPV=TP/(TP+FP)=(51+42)/16571). Higher PPV measurements are preferred since they indicate how likely a positive prediction is really an actual ncRNA. For example, by selecting a more stringent threshold of 0.75 as shown in Figure 3.20, we increased the PPV to 0.85% while maintaining an ncRNA gene sensitivity of 69%. These performance measurements assessed the ability of our classifier to identify ncRNA genes given all our positive predictions.

Additionally, we ran BLASTN on all our predicted candidates and found that over 95% of the hits had significant conservation (E-value ¡ $10^{-5}$) to other non-*E. coli* organisms, as shown in Figure 3.21. For the intergenic case, 93% predictions were conserved in other prokaryotic species, suggesting that these predictions were highly conserved independent of ORF conservation.

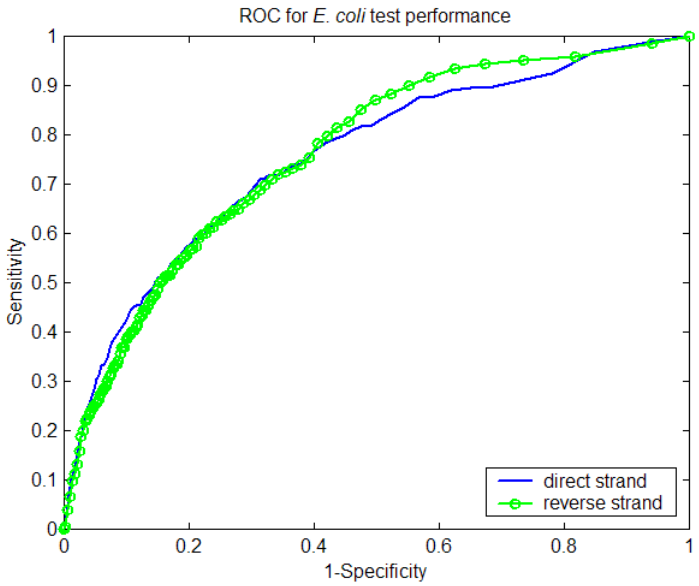


**Figure 3.19:** ROC curves for *E. coli* test performance on the direct and reverse strands.

### 3.2.5 Comparison of our prediction with other programs

We filtered our 16,571-predicted set using conservation, promoter, terminator, and positional information. The performance results presented here were strand independent since some existing programs [123, 153] gave *E. coli* ncRNA coordinates without specific strand information. Figure 3.22 shows a scatter plot of the sensitivity vs. PPV for various filtering combinations on our predictions compared to other programs in *E. coli*. As we can see from Figure 3.22, our prediction sensitivity and PPV values were comparable to or better than the best existing programs. Depending on the specific data available for filtering, we could identify a final set of candidates depending on whether preference is given towards choosing candidates with high sensitivity or high PPV.

Our *de novo* approach combined with specific filtering criteria outperformed three of the five existing programs as shown in Figure 3.22. In each of these cases, we could select specific filtering criteria that produce predictions with better sensitivity and PPV results than predictions by these programs. For illustration purposes, we selected a simple filtering strategy to balance the overall sensitivity and PPV that requires the ncRNA prediction to (1) fall into an antisense case and (2) have nucleotide overlap $< 50$ nt with a protein-coding

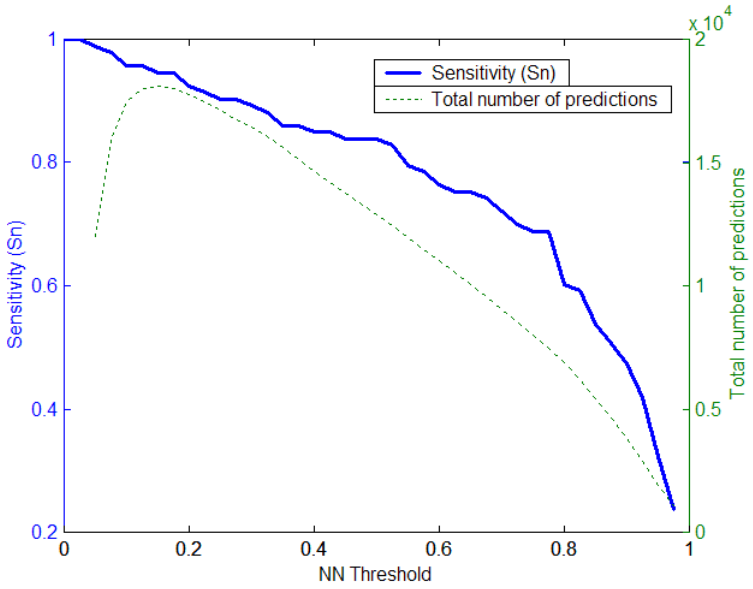


**Figure 3.20:** Sensitivity of known ncRNAs found (direct and reverse strands) versus NN threshold for *E. coli*. The total number of positive predictions versus NN threshold is also given. By adjusting the cutoff for the NN threshold, we can select the best trade-off in sensitivity and number of positive predictions.

region.

Based on the filtering criteria, we predicted 601 candidates and recovered 41% of known ncRNAs in *E. coli* with a PPV of 6%. Our list of 601 candidates included 23 candidates that overlapped known strand-specific ncRNAs, four candidates that overlapped annotated tRNAs, and 574 novel candidates. A summary of the prediction sensitivity (Sn) and positive prediction values (PPV) for the different programs is summarized in Table 3.7. Rivas et al. [123] had an overall better sensitivity and PPV than ours. However, their program relied on using prior knowledge of multiple alignments for identification of conserved regions, which may not be generally available for all genomes. Chen et al. [21] had better PPV but lower sensitivity than our program. Compared to Carter et al. [18], we had over 6% improvement in sensitivity with approximately equal PPV. Our predictions were also significantly better in Sn and PPV compared to Saetrom et al. [129] and Wang et al. [153].

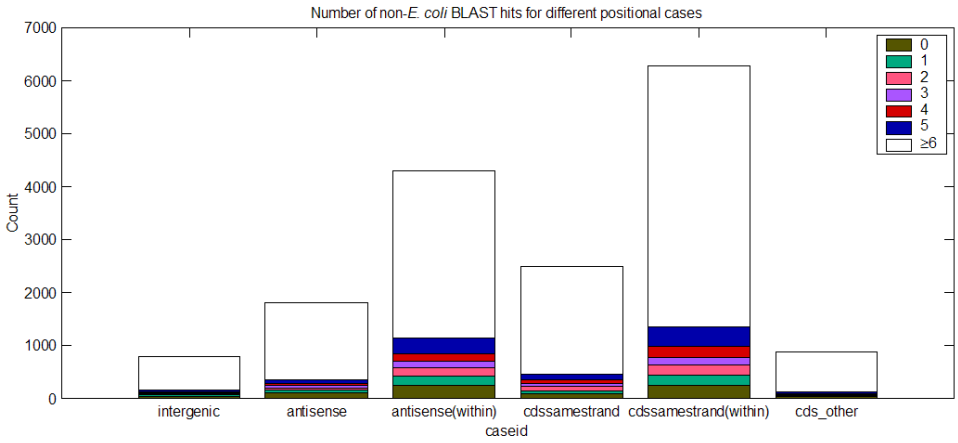


**Figure 3.21:** Number of non-*E. coli* BLAST hits for different positional cases: intergenic, antisense, cds_samestrand, and cds_other. Antisense and cds_samestrand cases are further subcategorized into those that partially or fully overlap (within) a protein-coding region. Significant BLAST hits were found in over 95% of our total predictions.

**Table 3.7:** Comparison of prediction performance by different programs. The number of predictions, sensitivity (Sn=TP/(TP+FN)), and positive prediction value (PPV=TP/(TP+FP)) is given for each program [18, 21, 122, 129, 153].

| Program | # predictions | Sn | PPV |
|---|---|---|---|
| Carter | 563 | 0.3441 | 0.0568 |
| Chen | 227 | 0.2903 | 0.1189 |
| Rivas | 275 | 0.4086 | 0.1382 |
| Saestrom | 306 | 0.1183 | 0.0359 |
| Wang | 420 | 0.0753 | 0.0167 |
| Tran | 601 | 0.4086 | 0.0632 |

#### *3.2.5.1 Experimental verification of selected ncRNA candidates*

By applying a specific set of filtering criteria based on conservation, promoter/terminator, positional, and tiling array data, we predicted 31 candidates for further validation, of which 17 overlapped with known ncRNA genes or annotated tRNA/rRNA genes in *E. coli*. From the 14 remaining novel predictions, eight were excluded because they overlapped with predicted ncRNA genes derived from other programs [18, 21, 122, 129, 153, 144]. The remaining six candidates (#5, 6, 8, 9, 11 and 12), as shown in Table 3.8, did not overlap with predictions by the other prediction programs, and had higher steady-state levels in the RNase E mutant. Based on our Northern analysis, three of the candidates (5, 6, and 8) were not observed in either the RNase E mutant or the wild type control (data not shown). Since

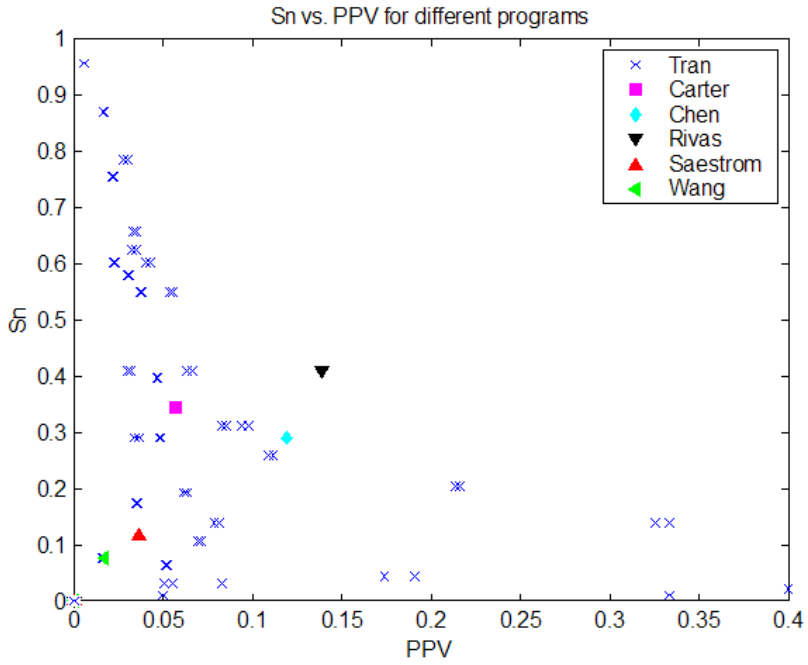


**Figure 3.22:** Comparison of performance for different programs in *E. coli*. The sensitivity (Sn=TP/(TP+FN)) and positive predictive value (PPV=TP/(TP+FP)) is shown for each program [18, 21, 122, 129, 153].

the tiling array has a higher sensitivity than the Northern analysis, we suspect that these potential ncRNAs are transcribed at such low levels that they could not be detected even in the RNase E deletion mutant.

Candidate 9 overlaps a region downstream of the rho-independent transcription terminator associated with the ydgA gene. It also overlaps a repetitive extragenic palindrome called RIP126 [127]. Using an RNA probe of 130 nt, a large species of 480 nt was observed in mid-log phase cells in a wild type strain (data not shown). Interestingly, significant

**Table 3.8:** Coordinates for the six ncRNA candidates chosen for experimental validation. The id, strand (0=direct, 1=reverse), start, and stop positions are given for each candidate.

| candidate # | strand | start | stop |
|---|---|---|---|
| 5 | 0 | 3086121 | 3086340 |
| 6 | 0 | 4213201 | 4213520 |
| 8 | 0 | 770261 | 770500 |
| 9 | 0 | 1689401 | 1689600 |
| 11 | 1 | 3399181 | 3399240 |
| 12 | 0 | 3483881 | 3484140 |

amounts of smaller species of 140, 170, and 215 nt appeared as the cells entered stationary phase (data not shown). However, there is considerable nucleotide sequence conservation among the various RIP elements; hence we designed a second RNA probe (a 30-mer) that was specific for RIP126. With this probe, we observed only the 480 nt species, which was visible in both mid-log and early stationary phase cells (data not shown). While we cannot conclude at this time if this species is a true ncRNA, we believe that it either represents a stable decay intermediate of the upstream ydgA mRNA or, if independently transcribed, contains a significant region of antisense to the 3' terminus of the adjacent uldC mRNA, which is transcribed in the opposite direction. It should also be noted that at least some of the RIP elements accumulate in stationary phase cells (data not shown).

Candidate 12 is located in the 5' untranslated region (UTR) of the crp gene. Previous experiments have shown the existence of three potential promoters (P1, P2, and P3) for this gene [70]. Transcription initiation from P3 would generate a 5' UTR of 167 nt. The RNA probe used was 89 nt in length and would detect RNA species arising from all three promoters. As shown in Figure 3.23A, a large number of discrete species were detected in the exponentially growing wild type cells, but most of them rapidly disappeared as the cells entered a stationary phase (data not shown). Strikingly, in the RNase E deletion the ∼700 nt transcript was the predominate species, demonstrating that almost all of the smaller products observed in the wild type control arose from RNase E cleavages. An ∼150 nt species was still detected in the RNase E mutant, which could have arisen from inefficient cleavages by RNase G. Since the large species detected in both the RNase E mutant and the wild type control was of the approximate size of the full-length crp mRNA, we speculate that all of the species observed in wild type cells (Figure 3.23A) are relatively stable mRNA decay products that retain some or all of the 5' UTR.

Candidate 11 falls within the 5' UTR of the mreB gene, a locus that is involved in establishment of the rod shape of the cell [137]. Transcriptional analysis of this gene has identified three potential promoters based on primer extension analysis [152]. Transcription from the most distal promoter would generate a 5' UTR of 267 nt. Using an RNA probe of 145 nt, we detected numerous species in the exponentially growing wild type cells (Figure

3.23B). Surprisingly, in the RNase E deletion strain, only a single 215 nt species was detected. Examination of the nucleotide sequence indicated that the potential P2 promoter observed by Wachi et al. [152] may represent an RNase E cleavage site (Figure 3.23C). In addition, when the 5' UTR was folded using RNAstar program, we observed a highly structured molecule (Figure 3.23C). While we cannot rule out at this time that there is a transcription termination site at the downstream stem-loop shown in Figure 3.23C, we believe that the there is a strong possibility that this candidate represents a riboswitch that helps regulate the expression of the mreBCD operon.

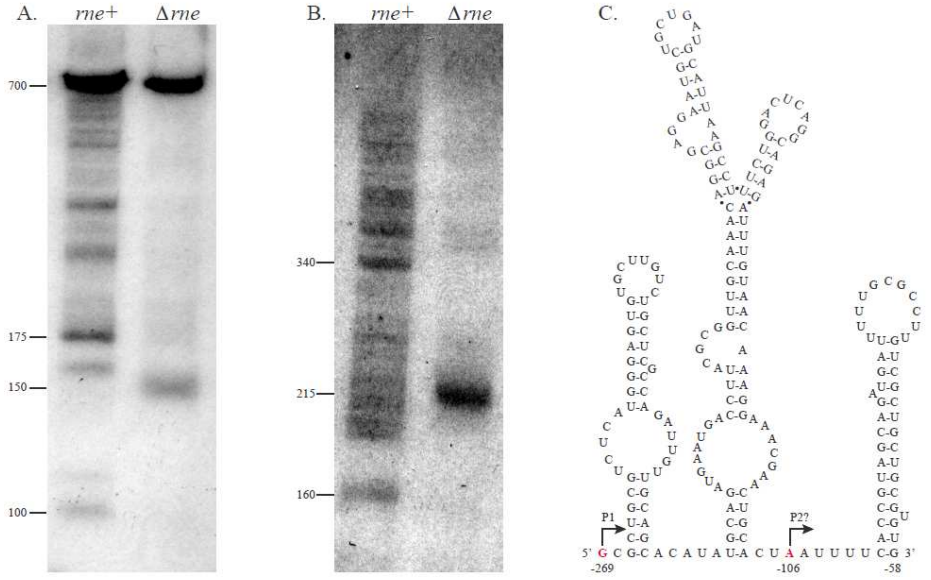


**Figure 3.23:** Analysis of predicted ncRNA candidates 11 and 12. (A) Northern analysis of candidate 12. Thirty $\mu$g of total RNA from exponentially growing MG1655 (*rne*+) and SK3564 ($\Delta$*rne*) was separated on a 6% PAGE as described in the Materials and Methods. Transcript sizes were estimated from a New England Biolabs low range ssRNA ladder. (B) Northern analysis of candidate 11. Thirty $\mu$g of total RNA from exponentially growing MG1655 (*rne*+) and SK3564 ($\Delta$*rne*) was separated on a 8% PAGE as described in the Materials and Methods. Transcript sizes were estimated from a New England Biolabs low range ssRNA ladder. (C) RNAstar secondary structure prediction of a portion of the mreB leader (nucleotides -269 to -58). Nucleotides shown in red at positions -269 and -106 correspond to the primer extension products detected by Wachi et al. [152]. Position -106 was originally identified as a potential transcription start site but probably represents an RNase E cleavage site.

### *3.3 Summary*

In this study, we identified a number of sequence and structure-based features that can distinguish known ncRNAs from their di-shuffled versions, which do not rely on a priori knowledge of sequence alignments, conservation with closely related organisms, or structural conservation. By utilizing these novel features, we developed a classifier for ncRNA gene prediction. The use of training samples from a large class of ncRNAs from diverse organisms enabled us to find different categories of ncRNAs from various organisms. Our program allows a user to include any prior knowledge about the target ncRNA genes while using our prediction program. For example, if a user has knowledge about the approximate length of ncRNAs, he/she can set the prediction window size accordingly, which could possibly result in an improved prediction, as we have demonstrated with *E. coli* in this study. In addition, specific filtering strategies can be customized based on the organism under investigation. For example, for some organisms, data from transcriptional signals such as promoters and terminators may not be readily available. In that case, one can use other data to narrow down the list of candidates after prediction.

Application of our program has led to a number of novel ncRNA gene predictions. Using Northern blot analysis, we were able to find expression in three out of six target candidates under our tested conditions. We believe the expressed candidates are stable decay products and one has the potential to be a riboswitch. Further functional experimental studies will be needed in order to fully verify these as real ncRNAs since transcription does not imply function.

The results of our ncRNA prediction in *E. coli* are shown to be highly competitive with or better than the existing prediction programs as we have well demonstrated in this study. Overall our genome-scale prediction results indicate that there may be many more ncRNAs in *E. coli*, particularly in non-intergenic regions, which have been missed by previous studies. Further functional studies on these predicted ncRNA genes are needed to better understand its role and mechanism in regulation.

# CHAPTER IV

# APPLICATIONS OF NON-CODING RNA PREDICTION

The purpose of this chapter is to investigate applications related to ncRNA gene prediction.

## *4.1 Relationship between operon structure and ncRNA gene prediction*

We explore the location of ncRNAs in relation to the genomic structure of an organism by examining higher-order gene organization, namely, operons, the basic transcriptional unit in prokaryotes. Very little is known about the transcriptional mechanism for ncRNAs. Besides being transcribed independently through promoter and terminator signals, ncRNAs are also believed to be processed from messenger RNA (mRNA) [31]. For example, in some small nucleolar RNAs (snoRNAs), the transcription of the ncRNA is initiated from the host gene's promoter [31]. Previous methods have relied on the assumption that ncRNAs are independently transcribed and have found ncRNAs by searching for short regions with promoter and terminator signals [6, 21]. In this study, we investigate whether operon information can be used to facilitate the identification of ncRNAs and find those potentially processed from a larger mRNA transcript.

We have examined where known ncRNAs are arranged in relation to operons. We divide the ncRNA arrangements into four different cases, as shown in Figure 4.1. Case1 occurs when an ncRNA does not overlap any ORF and therefore does not overlap any operon. Case2 is subdivided into case2(s) and case2(m) depending on whether the overlapping ORF belongs to a single gene operon or multiple gene operon, respectively. Case3 illustrates the condition when the ncRNA lies within an operon but does not overlap any ORFs. Case3 is an interesting situation where perhaps the operon's promoter initiates the transcription of the ncRNA gene.

Using our operon prediction results from [145], we examined the arrangement of 93 known ncRNAs in *E. coli*. The results are summarized in Table 4.1 for ncRNAs falling into the four different categories with and without regard to strand direction. Many ncRNAs

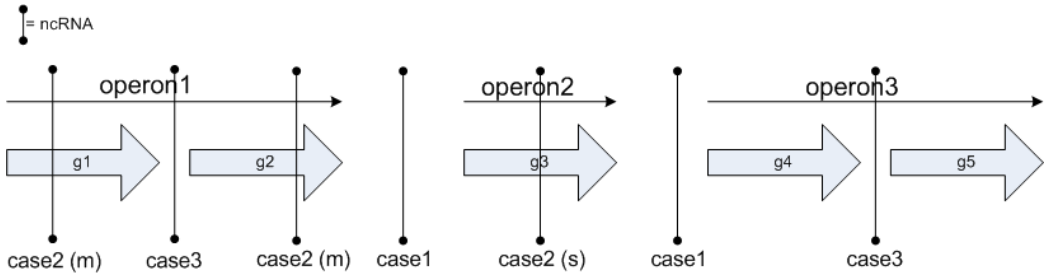


**Figure 4.1:** Schematic of different ncRNA arrangements in relation to operon structure.

**Table 4.1:** Percentage of ncRNAs cases with respect to operon structure for 93 known ncRNAs in *E. coli*.

| *E. coli* ncRNA-operon case | **with** regard to strand direction | **without** regard to strand direction |
|---|---|---|
| case1 | 77.42% | 51.61% |
| case2(s) | 10.75% | 36.56% |
| case2(m) | 11.83% | 11.83% |
| case3 | 0 | 0 |

belonging to case1 when strand direction was considered actually was recategorized into case2(s) when strand direction was not considered. This result is illustrated by the increase in the percentage of ncRNAs in case2(s) contained in the without regard to strand direction column of Table 4.1. Unfortunately, no examples of case3 were found in *E. coli*. The lack of case3 in *E. coli* may be due to various reasons, including the bias for genes within an operon to have shorter intergenic distances [131] and other reasons which we will discuss later.

We also investigated the ncRNA and operon genomic arrangement for other organisms using the 800 known ncRNAs across 194 different organisms introduced in Section 3.2.3. We used all available operon prediction results from [145, 161, 120]. The results of the four different categories with and without regard to strand direction are given in Table 4.2. This analysis also found similar results to *E. coli* since most of the ncRNAs fall into case1. It is interesting that up to 7% of known ncRNAs belong to case3 and have the potential to be processed from mRNA transcripts. An examination of the list of ncRNAs categorized as case3 indicates that the majority have lengths 40-60 nt. These ncRNAs are mostly annotated as snoRNAs.

The relatively low abundance of known ncRNAs falling into case3 may be the result of

**Table 4.2:** Percentage of ncRNAs cases with respect to operon structure for the dataset of 800 ncRNAs.

| 800 ncRNAs-operon case | **with** regard to strand direction | **without** regard to strand direction |
|---|---|---|
| case1 | 77.00% | 65.38% |
| case2(s) | 7.25% | 18.88% |
| case2(m) | 8.75% | 9.13% |
| case3 | 7% | 6.63% |

several factors. Operon prediction methods, including ours [145], apply intergenic distance as a feature used in prediction. The constraint on the intergenic distance also imposes a limitation on the type of ncRNAs that can be found. In our case, most of those found in case3 corresponded to snoRNAs. The typical lengths of snoRNAs (40-60 nt) actually favors their neighboring ORFs to be classified as a "within operon" gene pair. Additionally, current methods for finding ncRNAs are biased to finding candidates with independent transcription so the existing data set may have fewer examples of those transcribed together with other operons.

Based on the results of the ncRNA operon arrangement analysis, the use of operon prediction results do not seem to aid the general identification of ncRNAs. Perhaps only snoRNAs may potentially benefit from the additional analysis with operons. The use of operon prediction is recommended for further investigation in higher-level organisms with both operon structure and the presence of multiple mechanisms for mRNA processing, such as alternative splicing. Such organisms include the nematode, *Caenorhabditis elegans*, where various ncRNA transcriptional mechanisms have been proposed [31].

## *4.2 Application of ncRNA predictor to find ncRNAs in Sulfolobus solfataricus*

We illustrate the robustness of our ncRNA predictor to search for ncRNAs in the thermophilic archaeon, *Sulfolobus solfataricus* (NC_002754). The *Sulfolobus* species inhabit volcanic springs and geothermal areas of 75-80°C in an optimal pH of 2-3. The study of DNA replication in Archaea such as *Sulfolobus* are of interest since the mechanisms are evolutionarily conserved to that of Eukaryotes. Furthermore, the thermostability of proteins in *Sulfolobus* organisms has made them useful model organisms in wet labs. Recently, Tang

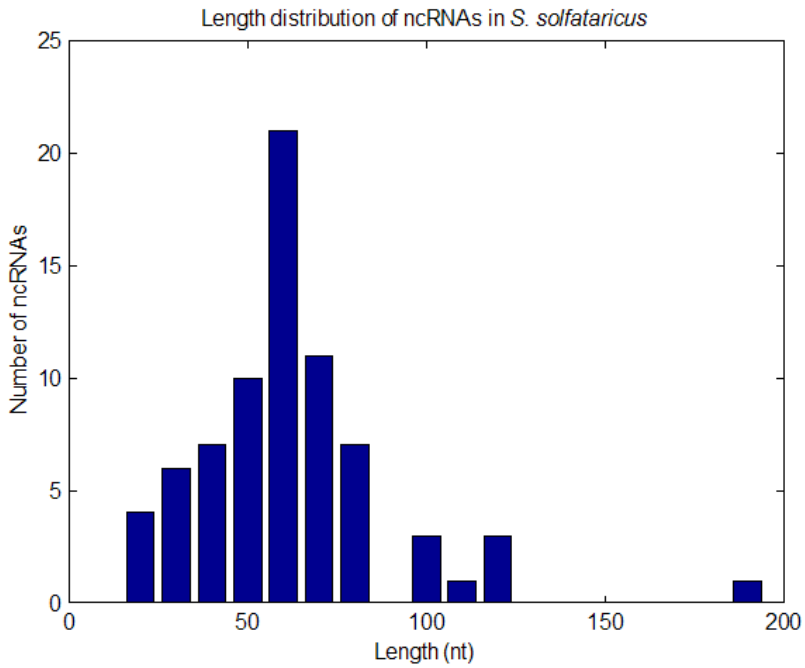


**Figure 4.2:** Histogram plot of the number of ncRNAs at each length contained in the known *S. solfataricus* set.

et al. [142] have used specialized cDNA library from *S. solfataricus* to identify 57 novel ncRNA candidates and confirmed their expression by Northern blot analysis. We chose to perform genome-wide prediction of ncRNAs using our predictor to assess the performance against all the known ncRNA genes in *S. solfataricus*. We have extracted all 74 known ncRNAs for *S. solfataricus* from the Positive1540 data set. This comprehensive list of ncRNAs includes all known candidates from [142], NONCODE, and GenBank. Using these ncRNAs as queries, we ran an all-versus-all BLASTN search against the Positive936 data set and removed all the Positive936 hits below an E-value cutoff of $10^{-5}$. We reduced the original Positive936 data set to 864 unique ncRNAs after removing sequences homologous to the 74 known ncRNAs. We use this data set without known ncRNAs in *S. solfataricus* for training and refer to it as Positive864_Sso. The length distribution for the ncRNAs in *S. solfataricus* is shown in Figure 4.2.

We used the same methodology used previously for *E. coli* in generating the negative set, which di-shuffles randomly selected sequence segments in *S. solfataricus* to ensure that no ncRNA-related secondary structure is present. The results from our ncRNA predictor

**Table 4.3:** AUROC performance of different window sizes and top number of features for direct and reverse (rc) strands.

| NC_002754 | f10 | f15 | f20 | f25 | f30 | f45 |
|---|---|---|---|---|---|---|
| test: | AUROC | AUROC | AUROC | AUROC | AUROC | AUROC |
| 40 | 0.7270 | 0.7223 | 0.7265 | 0.7405 | 0.7403 | 0.7388 |
| 40_rc | 0.7092 | 0.6987 | 0.7062 | 0.7114 | 0.7097 | 0.7088 |
| 80 | 0.6994 | 0.6990 | 0.7163 | 0.7074 | 0.7141 | 0.7090 |
| 80_rc | 0.6859 | 0.6682 | 0.6800 | 0.6739 | 0.6809 | 0.6673 |
| 120 | 0.6675 | 0.6608 | 0.6685 | 0.6788 | 0.6780 | 0.6582 |
| 120_rc | 0.6803 | 0.6625 | 0.6616 | 0.6679 | 0.6656 | 0.6371 |
| 160 | 0.6651 | 0.6476 | 0.6380 | 0.6338 | 0.6682 | 0.6265 |
| 160_rc | 0.6837 | 0.6391 | 0.6430 | 0.6138 | 0.6109 | 0.4632 |
| 200 | 0.6766 | 0.5738 | 0.5525 | 0.5731 | 0.6016 | 0.5500 |
| 200_rc | 0.6801 | 0.6211 | 0.5399 | 0.5486 | 0.5466 | 0.5042 |
| 240 | 0.6393 | 0.5913 | 0.5783 | 0.6111 | 0.5608 | 0.5932 |
| 240_rc | 0.6578 | 0.5398 | 0.5462 | 0.5740 | 0.5544 | 0.5424 |
| 280 | 0.6035 | 0.5202 | 0.5166 | 0.5184 | 0.5330 | 0.5136 |
| 280_rc | 0.6031 | 0.5351 | 0.5213 | 0.5122 | 0.5339 | 0.5000 |
| average | 0.6699 | 0.6271 | 0.6211 | 0.6261 | 0.6284 | 0.6009 |

are summarized in Table 4.3. The best average performance for *S. solfataricus* is obtained by using the top 10 features. The AUROC decreases for higher window sizes because fewer training samples are available and the neural network classifier tends to overfit the training data leading to lower generalization power in the test set. Examining the top 10 features for the window size $w = 40$ yields the following discriminative features (in descending order): diversity_free_energy_thermo_ensemble, kmer_CG, structuralstatistics_mfe, kmer_G, kmer_GC, kmer_UA, kmer_U, kmer_AU, rnacluster_ave_num_hifreq_bp_percluster, kmer_GG.

By applying the meta-learner to combine the prediction from different window sizes, we are able to obtain the final prediction results shown in Table 4.4. We present the results using three window sizes, $w$=40, 80, 120 and the results using all seven standard window sizes $w$=40, 80, 120, 160, 200, 240, 280. Similar to *E. coli*, we observe higher performance when we use typical window sizes corresponding to the lengths of the ncRNAs found in the organism.

**Table 4.4:** AUROC performance of ncRNA meta-learner predictor for direct and reverse strands using different number of features.

| # win sizes | Strand | f10 | f15 | f20 | f25 | f30 | f45 |
|---|---|---|---|---|---|---|---|
| 3 | Direct | 0.7442 | 0.7284 | 0.7573 | 0.7623 | 0.7628 | 0.7502 |
| 3 | Reverse | 0.7296 | 0.7072 | 0.7243 | 0.7266 | 0.7308 | 0.7214 |
| 7 | Direct | 0.6848 | 0.6161 | 0.5969 | 0.5859 | 0.5797 | 0.5429 |
| 7 | Reverse | 0.6448 | 0.6077 | 0.5650 | 0.5552 | 0.5515 | 0.5111 |

The application of our ncRNA meta-learner predictor to *S. solfataricus* is highly promising. We demonstrate that by selecting the top three window sizes corresponding to typical ncRNA lengths found in the organism, we can obtain AUROC above 0.7 for predicting windows with potential for ncRNAs. Using the standard window sizes gives somewhat lower performance; however, we can still obtain AUROC above 0.64 if we limit the number of features used to help the classifier generalize better to the test set.

# CHAPTER V

# CONCLUSIONS

In this dissertation, we applied pattern recognition approaches from signal processing to research in the area of genome annotation. We demonstrated the use of meta-learner classifiers in both operon prediction and ncRNA gene prediction. We have confirmed the results of the predictors by comparing the performance to existing algorithms and applied experimental validation when available.

## *5.1 Contributions*

We have presented results in the following two main areas:

- Our work on operon prediction demonstrates how we can improve upon the accuracy of existing methods using a meta-learning approach. We have successfully applied our novel method to predict operons in the bacteria *Escherichia coli* and *Bacillus subtilis* and the hyperthermophilic archaeon *Pyrococcus furiosus*. Our operon predictions show significant improvement in performance above existing methods. The ability to predict operons allows for the functional inference of hypothetical and conserved hypothetical genes, and represents a key step in reconstructing biological pathways and networks for prokaryotes.

- We developed a *de novo* computational method for the prediction of ncRNAs in prokaryotes. Our work on non-coding RNA gene prediction enables the identification of both *cis*- and *trans*- regulating ncRNAs without limiting the search space and requiring prior knowledge of sequence alignments, conservation with closely related organisms, or structural conservation. We believe that this work will contribute to a more comprehensive annotation of ncRNAs in partially and completely sequenced microbial genomes.

## *5.2 Publications*

The work related to this dissertation has led to the following list of publications:

**Journals**

1. **T. T. Tran**, F. Zhou, S. Marshburn, M. Stead, S. R. Kushner, and Y. Xu, "*De novo* computational prediction of non-coding RNA genes in prokaryotic genomes," submitted, 2009.

2. **T. T. Tran**, P. Dam, Z. Su, F. L. Poole II, M. W. Adams, G. T. Zhou, and Y. Xu, "Operon prediction in *Pyrococcus furiosus*," *Nucleic Acids Res*, vol. 35, no. 1, pp. 11-20, 2007.

3. F. Zhou, **T. Tran**, and Y. Xu, "*Nezha*, a novel active miniature inverted-repeat transposable element in cyanobacteria," *Biochem Biophys Res Commun*, vol. 365, pp. 790-4, Jan 25 2008.

**Books**

4. P. Dam, F. Mao, D. Che, P. Wan, **T. Tran**, G. Li, and Y. Xu, "Computational elucidation of operons and uber-operons," in *Computational Methods for Understanding Bacterial and Archaeal Genomes*, vol. 7, Y. Xu and J. P. Gogarten, Eds.: Imperial College Press, 2008.

**Conferences**

5. **T. T. Tran**, V. A. Emanuele II, and G. T. Zhou, "Techniques for detecting approximate tandem repeats in DNA," *Proc. IEEE Intl. Conference on Acoustics, Speech, and Signal Processing*, pp. 449-452, Montreal, Canada, May 2004.

6. V. A. Emanuele II, **T. T. Tran**, and G. T. Zhou, "A Fourier product method for detecting approximate tandem repeats in DNA," *Proc. IEEE Workshop on Statistical Signal Processing*, Bordeaux, France, July 2005.

7. **T. T. Tran**, F. Poole, G. T. Zhou, and Y. Xu, "An integrative approach to operon prediction in *Pyrococcus furiosus*," Poster presentation at the First Annual Symposium on Computational and Systems Biology, Athens, Georgia, USA, November 11, 2005.

8. **T. T. Tran** and Y. Xu, "On the road to genomic annotation: from operons to non-coding RNAs," Poster presentation at the Second Annual Symposium on Computational and Systems Biology, Athens, Georgia, USA, March 23, 2007.

9. **T. T. Tran**, Q. Liu, and Y. Xu, "Investigation of RNA Secondary Structure Clustering for the Identification of Small Non-coding RNAs," Poster presentation at the Third Annual Symposium on Computational and Systems Biology, Athens, Georgia, USA, March 21, 2008.

### *5.3 Future Work*

The following represents a list of interesting research topics for future investigation:

- Investigate the functions of structural non-coding RNAs by identifying their possible messenger RNA (mRNA) targets. Examine functional Gene Ontology (GO) and KEGG Orthology (KO) significance of mRNA targets.

- Incorporate the operon prediction and ncRNA gene prediction into an automated pipeline for large-scale genome annotation. Automate a web-server to update the training set and compute predictions accordingly from the latest releases of ncRNA databases and GenBank annotation. Create a user-friendly web interface to allow users to submit partial or completely sequenced genomes for ncRNA gene annotation.

- Perform large scale ncRNA gene prediction on various groups of organisms, including *Cyanobacteria*, to study the role of ncRNAs in regulation and their ability to enable the organism to adapt and thrive in extreme environments.

- Examine the role of ncRNAs in specific regulatory and metabolic pathways for its application to alternative energy sources such as biofuels.

- Apply our ncRNA gene prediction method to identify human microRNAs (miRNA) and small interfering RNAs (siRNAs). Investigate the role of ncRNAs in complex diseases such as cancer.

# REFERENCES


[1] "Part c: Hyperthermophilic enzymes," in *Part C: Hyperthermophilic Enzymes* (Adams, M. W. W. and Kelly, R. M., eds.), vol. 334 of *Methods in Enzymology*, pp. 3–526, New York: Methods in Enzymology, Academic Press, 2001.

[2] Allers, J. and Shamoo, Y., "Structure-based analysis of protein-rna interactions using the program entangle," *J Mol Biol*, vol. 311, no. 1, pp. 75–86, 2001.

[3] Altschul, S. F. and Erickson, B. W., "Significance of nucleotide sequence alignments: a method for random sequence permutation that preserves dinucleotide and codon usage," *Mol Biol Evol*, vol. 2, no. 6, pp. 526–38, 1985.

[4] Altschul, S. F., Madden, T. L., Schaffer, A. A., Zhang, J., Zhang, Z., Miller, W., and Lipman, D. J., "Gapped blast and psi-blast: a new generation of protein database search programs," *Nucleic Acids Res*, vol. 25, no. 17, pp. 3389–402, 1997.

[5] Anastassiou, D., "Genomic signal processing," *IEEE Signal Processing Magazine*, vol. 18, no. 4, pp. 8–20, 2001.

[6] Argaman, L., Hershberg, R., Vogel, J., Bejerano, G., Wagner, E. G., Margalit, H., and Altuvia, S., "Novel small rna-encoding genes in the intergenic regions of *Escherichia coli*," *Curr Biol*, vol. 11, no. 12, pp. 941–50, 2001.

[7] Ashburner, M., Ball, C. A., Blake, J. A., Botstein, D., Butler, H., Cherry, J. M., Davis, A. P., Dolinski, K., Dwight, S. S., Eppig, J. T., Harris, M. A., Hill, D. P., Issel-Tarver, L., Kasarskis, A., Lewis, S., Matese, J. C., Richardson, J. E., Ringwald, M., Rubin, G. M., and Sherlock, G., "Gene ontology: tool for the unification of biology. the gene ontology consortium," *Nat Genet*, vol. 25, no. 1, pp. 25–9, 2000.

[8] Axmann, I. M., Kensche, P., Vogel, J., Kohl, S., Herzel, H., and Hess, W. R., "Identification of cyanobacterial non-coding rnas by comparative genome analysis," *Genome Biol*, vol. 6, no. 9, p. R73, 2005.

[9] Bachellerie, J. P., Cavaille, J., and Huttenhofer, A., "The expanding snorna world," *Biochimie*, vol. 84, no. 8, pp. 775–90, 2002.

[10] Bailey, T. L. and Elkan, C., "The value of prior knowledge in discovering motifs with meme," *Proc Int Conf Intell Syst Mol Biol*, vol. 3, pp. 21–9, 1995.

[11] Bernstein, J. A., Lin, P. H., Cohen, S. N., and Lin-Chao, S., "Global analysis of *Escherichia coli* rna degradosome function using dna microarrays," *Proc Natl Acad Sci U S A*, vol. 101, no. 9, pp. 2758–63, 2004.

[12] Bockhorst, J., Craven, M., Page, D., Shavlik, J., and Glasner, J., "A bayesian network approach to operon prediction," *Bioinformatics*, vol. 19, no. 10, pp. 1227–35, 2003.

[13] Bompfunewerer, A., Flamm, C., Fried, C., Fritzsch, G., Hofacker, I., Lehmann, J., Missal, K., Mosig, A., Muller, B., Prohaska, S., Stadler, B., Stadler, P., Tanzer, A., Washietl, S., and Witwer, C., "Evolutionary patterns of non-coding rnas," *Theory in Biosciences*, vol. 123, no. 4, pp. 301–369, 2005.

[14] Bonnet, E., Wuyts, J., Rouze, P., and Van de Peer, Y., "Evidence that microrna precursors, unlike other non-coding rnas, have lower folding free energies than random sequences," *Bioinformatics*, vol. 20, no. 17, pp. 2911–7, 2004.

[15] Brocchieri, L. and Karlin, S., "Protein length in eukaryotic and prokaryotic proteomes," *Nucleic Acids Res*, vol. 33, no. 10, pp. 3390–400, 2005.

[16] Buckingham, S., "The major world of microrna," in *Horizon Symposia Understanding the RNAIssance*, pp. 1–3, 2003.

[17] Cao, H. B., Wang, C. Z., Dobbs, D., Ihm, Y., and Ho, K. M., "Codability criterion for picking proteinlike structures from random three-dimensional configurations," *Phys Rev E Stat Nonlin Soft Matter Phys*, vol. 74, no. 3 Pt 1, p. 031921, 2006.

[18] Carter, R. J., Dubchak, I., and Holbrook, S. R., "A computational approach to identify genes for functional rnas in genomic sequences," *Nucleic Acids Res*, vol. 29, no. 19, pp. 3928–38, 2001.

[19] Carthew, R. W., "Molecular biology. a new rna dimension to genome control," *Science*, vol. 313, no. 5785, pp. 305–6, 2006.

[20] Chan, C. Y. and Ding, Y., "Boltzmann ensemble features of rna secondary structures: a comparative analysis of biological rna sequences and random shuffles," *J Math Biol*, vol. 56, no. 1-2, pp. 93–105, 2008.

[21] Chen, S., Lesnik, E. A., Hall, T. A., Sampath, R., Griffey, R. H., Ecker, D. J., and Blyn, L. B., "A bioinformatics based approach to discover small rna genes in the *Escherichia coli* genome," *Biosystems*, vol. 65, no. 2-3, pp. 157–77, 2002.

[22] Chen, X., Su, Z., Dam, P., Palenik, B., Xu, Y., and Jiang, T., "Operon prediction by comparative genomics: an application to the *Synechococcus* sp. wh8102 genome," *Nucleic Acids Res*, vol. 32, no. 7, pp. 2147–57, 2004.

[23] Chen, X., Su, Z., Xu, Y., and Jiang, T., "Computational prediction of operons in *Synechococcus* sp. wh8102," *Proceedings of 15th International Conference on Genome Informatics*, vol. 15, no. 2, pp. 211–22, 2004.

[24] Clote, P., "An efficient algorithm to compute the landscape of locally optimal rna secondary structures with respect to the nussinov-jacobson energy model," *J Comput Biol*, vol. 12, no. 1, pp. 83–101, 2005.

[25] Clote, P., Ferre, F., Kranakis, E., and Krizanc, D., "Structural rna has lower folding energy than random rna of the same dinucleotide frequency," *Rna*, vol. 11, no. 5, pp. 578–91, 2005.

[26] Coventry, A., Kleitman, D. J., and Berger, B., "Msari: multiple sequence alignments for statistical detection of rna secondary structure," *Proc Natl Acad Sci U S A*, vol. 101, no. 33, pp. 12102–7, 2004.

[27] Craven, M., Page, D., Shavlik, J., Bockhorst, J., and Glasner, J., "A probabilistic learning approach to whole-genome operon prediction," *Proc Int Conf Intell Syst Mol Biol*, vol. 8, pp. 116–27, 2000.

[28] Dam, P., Olman, V., Harris, K., Su, Z., and Xu, Y., "Operon prediction using both genome-specific and general genomic information," *Nucleic Acids Res*, vol. 35, no. 1, pp. 288–98, 2007.

[29] de Hoon, M. J., Imoto, S., Kobayashi, K., Ogasawara, N., and Miyano, S., "Predicting the operon structure of *Bacillus subtilis* using operon length, intergene distance, and gene expression information," *Pac Symp Biocomput*, pp. 276–87, 2004.

[30] de Hoon, M. J., Makita, Y., Nakai, K., and Miyano, S., "Prediction of transcriptional terminators in *Bacillus subtilis* and related species," *PLoS Comput Biol*, vol. 1, no. 3, p. e25, 2005.

[31] Deng, W., Zhu, X., Skogerbo, G., Zhao, Y., Fu, Z., Wang, Y., He, H., Cai, L., Sun, H., Liu, C., Li, B., Bai, B., Wang, J., Jia, D., Sun, S., He, H., Cui, Y., Wang, Y., Bu, D., and Chen, R., "Organization of the *Caenorhabditis elegans* small non-coding transcriptome: Genomic features, biogenesis, and expression," *Genome Res*, vol. 16, no. 1, pp. 20–9, 2006.

[32] di Bernardo, D., Down, T., and Hubbard, T., "ddbrna: detection of conserved secondary structures in multiple alignments," *Bioinformatics*, vol. 19, no. 13, pp. 1606–11, 2003.

[33] Ding, Y., Chan, C. Y., and Lawrence, C. E., "Rna secondary structure prediction by centroids in a boltzmann weighted ensemble," *Rna*, vol. 11, no. 8, pp. 1157–66, 2005.

[34] Ding, Y., Chan, C. Y., and Lawrence, C. E., "Clustering of rna secondary structures with application to messenger rnas," *J Mol Biol*, vol. 359, no. 3, pp. 554–71, 2006.

[35] Ding, Y. and Lawrence, C. E., "A statistical sampling algorithm for rna secondary structure prediction," *Nucleic Acids Res*, vol. 31, no. 24, pp. 7280–301, 2003.

[36] Dougherty, E. R. and Datta, A., "Genomic signal processing: diagnosis and therapy," *IEEE Signal Processing Magazine*, vol. 22, no. 1, pp. 107–12, 2005.

[37] Dougherty, E. R., Datta, A., and Sima, C., "Research issues in genomic signal processing," *IEEE Signal Processing Magazine*, vol. 22, no. 6, pp. 46–68, 2005.

[38] Draper, D. E., "Themes in rna-protein recognition," *J Mol Biol*, vol. 293, no. 2, pp. 255–70, 1999.

[39] Eddy, S. R., "Non-coding rna genes and the modern rna world," *Nat Rev Genet*, vol. 2, no. 12, pp. 919–29, 2001.

[40] Eddy, S. R., "Computational genomics of noncoding rna genes," *Cell*, vol. 109, no. 2, pp. 137–40, 2002.

[41] Eddy, S. R., "A memory-efficient dynamic programming algorithm for optimal alignment of a sequence to an rna secondary structure," *BMC Bioinformatics*, vol. 3, p. 18, 2002.

[42] Eddy, S. R., "How do rna folding algorithms work?," *Nat Biotechnol*, vol. 22, no. 11, pp. 1457–8, 2004.

[43] Edvardsson, S., Gardner, P. P., Poole, A. M., Hendy, M. D., Penny, D., and Moulton, V., "A search for h/aca snornas in yeast using mfe secondary structure prediction," *Bioinformatics*, vol. 19, no. 7, pp. 865–73, 2003.

[44] Edwards, M. T., Rison, S. C., Stoker, N. G., and Wernisch, L., "A universally applicable method of operon map prediction on minimally annotated genomes using conserved genomic context," *Nucleic Acids Res*, vol. 33, no. 10, pp. 3253–62, 2005.

[45] Enright, A. J., Van Dongen, S., and Ouzounis, C. A., "An efficient algorithm for large-scale detection of protein families," *Nucleic Acids Res*, vol. 30, no. 7, pp. 1575–84, 2002.

[46] Ermolaeva, M. D., White, O., and Salzberg, S. L., "Prediction of operons in microbial genomes," *Nucleic Acids Res*, vol. 29, no. 5, pp. 1216–21, 2001.

[47] Fawcett, T., "Roc graphs: Notes and practical considerations for researchers," tech. rep., HP Laboratories, 2004.

[48] Fiala, G. and Stetter, K. O., "*Pyrococcus furiosus* sp. nov. represents a novel genus of marine heterotrophic archaebacteria growing optimally at 100°C," *Archives of Microbiology*, vol. 145, no. 1, pp. 56–61, 1986.

[49] Fickett, J. W. and Tung, C. S., "Assessment of protein coding measures," *Nucleic Acids Res*, vol. 20, no. 24, pp. 6441–50, 1992.

[50] Fickett, J. W., "The gene identification problem: an overview for developers," *Computers Chem*, vol. 20, no. 1, pp. 103–118, 1996.

[51] Fire, A., Xu, S., Montgomery, M. K., Kostas, S. A., Driver, S. E., and Mello, C. C., "Potent and specific genetic interference by double-stranded rna in *Caenorhabditis elegans*," *Nature*, vol. 391, no. 6669, pp. 806–11, 1998.

[52] Freyhult, E., Gardner, P. P., and Moulton, V., "A comparison of rna folding measures," *BMC Bioinformatics*, vol. 6, p. 241, 2005.

[53] Freyhult, E., Moulton, V., and Clote, P., "Rnabor: a web server for rna structural neighbors," *Nucleic Acids Res*, vol. 35, no. Web Server issue, pp. W305–9, 2007.

[54] Ganapathiraju, M. K., Klein-Seetharaman, J., Balakrishnan, N., and Reddy, R., "Characterization of protein secondary structure," *IEEE Signal Processing Magazine*, vol. 21, no. 3, pp. 78–87, 2004.

[55] Gaspin, C., Cavaille, J., Erauso, G., and Bachellerie, J. P., "Archaeal homologs of eukaryotic methylation guide small nucleolar rnas: lessons from the *Pyrococcus* genomes," *J Mol Biol*, vol. 297, no. 4, pp. 895–906, 2000.

[56] Gautheret, D. and Lambert, A., "Direct rna motif definition and identification from multiple sequence alignments using secondary structure profiles," *J Mol Biol*, vol. 313, no. 5, pp. 1003–11, 2001.

[57] Gong, H., Liu, C. M., Liu, D. P., and Liang, C. C., "The role of small rnas in human diseases: potential troublemaker and therapeutic tools," *Med Res Rev*, vol. 25, no. 3, pp. 361–81, 2005.

[58] Gottesman, S., "Micros for microbes: non-coding regulatory rnas in bacteria," *Trends Genet*, vol. 21, no. 7, pp. 399–404, 2005.

[59] Griffiths-Jones, S., Bateman, A., Marshall, M., Khanna, A., and Eddy, S. R., "Rfam: an rna family database," *Nucleic Acids Res*, vol. 31, no. 1, pp. 439–41, 2003.

[60] Gruber, A. R., Lorenz, R., Bernhart, S. H., Neubock, R., and Hofacker, I. L., "The vienna rna websuite," *Nucleic Acids Res*, 2008.

[61] Harris, J. K., Haas, E. S., Williams, D., Frank, D. N., and Brown, J. W., "New insight into rnase p rna structure from comparative analysis of the archaeal rna," *Rna*, vol. 7, no. 2, pp. 220–32, 2001.

[62] Hertel, J., Lindemeyer, M., Missal, K., Fried, C., Tanzer, A., Flamm, C., Hofacker, I. L., and Stadler, P. F., "The expansion of the metazoan microrna repertoire," *BMC Genomics*, vol. 7, p. 25, 2006.

[63] Hiller, M., Pudimat, R., Busch, A., and Backofen, R., "Using rna secondary structures to guide sequence motif finding towards single-stranded regions," *Nucleic Acids Res*, vol. 34, no. 17, p. e117, 2006.

[64] Hinas, A. and Soderbom, F., "Treasure hunt in an amoeba: non-coding rnas in *Dictyostelium discoideum*," *Curr Genet*, vol. 51, no. 3, pp. 141–159, 2007.

[65] Hofacker, I. L., "Vienna rna secondary structure server," *Nucleic Acids Res*, vol. 31, no. 13, pp. 3429–31, 2003.

[66] Huttenhofer, A., Brosius, J., and Bachellerie, J. P., "Rnomics: identification and function of small, non-messenger rnas," *Curr Opin Chem Biol*, vol. 6, no. 6, pp. 835–43, 2002.

[67] Huttenhofer, A., Schattner, P., and Polacek, N., "Non-coding rnas: hope or hype?," *Trends Genet*, vol. 21, no. 5, pp. 289–97, 2005.

[68] Huttenhofer, A. and Vogel, J., "Experimental approaches to identify non-coding rnas," *Nucleic Acids Res*, vol. 34, no. 2, pp. 635–46, 2006.

[69] Huynen, M., Gutell, R., and Konings, D., "Assessing the reliability of rna folding using statistical mechanics," *J Mol Biol*, vol. 267, no. 5, pp. 1104–12, 1997.

[70] Ishizuka, H., Hanamura, A., Inada, T., and Aiba, H., "Mechanism of the down-regulation of camp receptor protein by glucose in *Escherichia coli* : role of autoregulation of the crp gene," *EMBO J*, vol. 13, no. 13, pp. 3077–82, 1994.

[71] Jacob, E., Sasikumar, R., and Nair, K. N., "A fuzzy guided genetic algorithm for operon prediction," *Bioinformatics*, vol. 21, no. 8, pp. 1403–7, 2005.

[72] Jaeger, J. A., Turner, D. H., and Zuker, M., "Improved predictions of secondary structures for rna," *Proc Natl Acad Sci U S A*, vol. 86, no. 20, pp. 7706–10, 1989.

[73] Janga, S. C. and Moreno-Hagelsieb, G., "Conservation of adjacency as evidence of paralogous operons," *Nucleic Acids Res*, vol. 32, no. 18, pp. 5392–7, 2004.

[74] Jie, C. and Wong, S. T. C., "Nanotechnology for genomic signal processing in cancer research: a focus on the genomic signal processing hardware design of the nanotools for cancer research," *IEEE Signal Processing Magazine*, vol. 24, no. 1, pp. 111–21, 2007.

[75] Jones, S., Daley, D. T., Luscombe, N. M., Berman, H. M., and Thornton, J. M., "Protein-rna interactions: a structural analysis," *Nucleic Acids Res*, vol. 29, no. 4, pp. 943–54, 2001.

[76] Jones-Rhoades, M. W. and Bartel, D. P., "Computational identification of plant micrornas and their targets, including a stress-induced mirna," *Mol Cell*, vol. 14, no. 6, pp. 787–99, 2004.

[77] Kanehisa, M., Goto, S., Hattori, M., Aoki-Kinoshita, K. F., Itoh, M., Kawashima, S., Katayama, T., Araki, M., and Hirakawa, M., "From genomics to chemical genomics: new developments in kegg," *Nucleic Acids Res*, vol. 34, no. Database issue, pp. D354–7, 2006.

[78] Kawano, M., Reynolds, A. A., Miranda-Rios, J., and Storz, G., "Detection of 5'- and 3'-utr-derived small rnas and cis-encoded antisense rnas in *Escherichia coli*," *Nucleic Acids Res*, vol. 33, no. 3, pp. 1040–50, 2005.

[79] Kingsford, C. L., Ayanbule, K., and Salzberg, S. L., "Rapid, accurate, computational discovery of rho-independent transcription terminators illuminates their relationship to dna uptake," *Genome Biol*, vol. 8, no. 2, p. R22, 2007.

[80] Klein, R. J. and Eddy, S. R., "Rsearch: finding homologs of single structured rna sequences," *BMC Bioinformatics*, vol. 4, p. 44, 2003.

[81] Klein, R. J., Misulovin, Z., and Eddy, S. R., "Noncoding rna genes identified in at-rich hyperthermophiles," *Proc Natl Acad Sci U S A*, vol. 99, no. 11, pp. 7542–7, 2002.

[82] Kletzin, A. and Adams, M. W., "Molecular and phylogenetic characterization of pyruvate and 2-ketoisovalerate ferredoxin oxidoreductases from *Pyrococcus furiosus* and pyruvate ferredoxin oxidoreductase from *Thermotoga maritima*," *J Bacteriol*, vol. 178, no. 1, pp. 248–57, 1996.

[83] Kulkarni, P. R., Cui, X., Williams, J. W., Stevens, A. M., and Kulkarni, R. V., "Prediction of csra-regulating small rnas in bacteria and their experimental verification in *Vibrio fischeri*," *Nucleic Acids Res*, vol. 34, no. 11, pp. 3361–9, 2006.

[84] Lagesen, K., Hallin, P., Rodland, E. A., Staerfeldt, H. H., Rognes, T., and Ussery, D. W., "Rnammer: consistent and rapid annotation of ribosomal rna genes," *Nucleic Acids Res*, vol. 35, no. 9, pp. 3100–8, 2007.

[85] Lai, E. C., Tomancak, P., Williams, R. W., and Rubin, G. M., "Computational identification of *Drosophila* microrna genes," *Genome Biol*, vol. 4, no. 7, p. R42, 2003.

[86] Lander, E. S., Linton, L. M., Birren, B., Nusbaum, C., Zody, M. C., Baldwin, J., Devon, K., Dewar, K., Doyle, M., FitzHugh, W., Funke, R., Gage, D., Harris, K., Heaford, A., Howland, J., Kann, L., Lehoczky, J., LeVine, R., McEwan, P., McKernan, K., Meldrim, J., Mesirov, J. P., Miranda, C., Morris, W., Naylor, J., Raymond, C., Rosetti, M., Santos, R., Sheridan, A., Sougnez, C., Stange-Thomann, N., Stojanovic, N., Subramanian, A., Wyman, D., Rogers, J., Sulston, J., Ainscough, R., Beck, S., Bentley, D., Burton, J., Clee, C., Carter, N., Coulson, A., Deadman, R., Deloukas, P., Dunham, A., Dunham, I., Durbin, R., French, L., Grafham, D., Gregory, S., Hubbard, T., Humphray, S., Hunt, A., Jones, M., Lloyd, C., McMurray, A., Matthews, L., Mercer, S., Milne, S., Mullikin, J. C., Mungall, A., Plumb, R., Ross, M., Shownkeen, R., Sims, S., Waterston, R. H., Wilson, R. K., Hillier, L. W., McPherson, J. D., Marra, M. A., Mardis, E. R., Fulton, L. A., Chinwalla, A. T., Pepin, K. H., Gish, W. R., Chissoe, S. L., Wendl, M. C., Delehaunty, K. D., Miner, T. L., Delehaunty, A., Kramer, J. B., Cook, L. L., Fulton, R. S., Johnson, D. L., Minx, P. J., Clifton, S. W., Hawkins, T., Branscomb, E., Predki, P., Richardson, P., Wenning, S., Slezak, T., Doggett, N., Cheng, J. F., Olsen, A., Lucas, S., Elkin, C., Uberbacher, E., Frazier, M., and others, "Initial sequencing and analysis of the human genome," *Nature*, vol. 409, no. 6822, pp. 860–921, 2001.

[87] Larsson, P., Hinas, A., Ardell, D. H., Kirsebom, L. A., Virtanen, A., and Soderbom, F., "De novo search for non-coding rna genes in the at-rich genome of dictyostelium discoideum: performance of markov-dependent genome feature scoring," *Genome Res*, vol. 18, no. 6, pp. 888–99, 2008.

[88] Laslett, D., Canback, B., and Andersson, S., "Bruce: a program for the detection of transfer-messenger rna genes in nucleotide sequences," *Nucleic Acids Res*, vol. 30, no. 15, pp. 3449–53, 2002.

[89] Liang, M. P., Troyanskaya, O. G., Laederach, A., Brutlag, D. L., and Altman, R. B., "Computational functional genomics," *IEEE Signal Processing Magazine*, vol. 21, no. 6, pp. 62–9, 2004.

[90] Lim, L. P., Glasner, M. E., Yekta, S., Burge, C. B., and Bartel, D. P., "Vertebrate microrna genes," *Science*, vol. 299, no. 5612, p. 1540, 2003.

[91] Lim, L. P., Lau, N. C., Weinstein, E. G., Abdelhakim, A., Yekta, S., Rhoades, M. W., Burge, C. B., and Bartel, D. P., "The microrna of *Caenorhabditis elegans*," *Genes Dev*, vol. 17, no. 8, pp. 991–1008, 2003.

[92] Liolios, K., Tavernarakis, N., Hugenholtz, P., and Kyrpides, N. C., "The genomes on line database (gold) v.2: a monitor of genome projects worldwide," *Nucleic Acids Res*, vol. 34, no. Database issue, pp. D332–4, 2006.

[93] Liu, C., Bai, B., Skogerbo, G., Cai, L., Deng, W., Zhang, Y., Bu, D., Zhao, Y., and Chen, R., "Noncode: an integrated knowledge database of non-coding rnas," *Nucleic Acids Res*, vol. 33, no. Database issue, pp. D112–5, 2005.

[94] Liu, J., Gough, J., and Rost, B., "Distinguishing protein-coding from non-coding rnas through support vector machines," *PLoS Genet*, vol. 2, no. 4, p. e29, 2006.

[95] Liu, Q., Olman, V., Liu, H., Ye, X., Qiu, S., and Xu, Y., "Rnacluster: An integrated tool for rna secondary structure comparison and clustering," *J Comput Chem*, vol. 29, no. 9, pp. 1517–26, 2008.

[96] Livny, J., Brencic, A., Lory, S., and Waldor, M. K., "Identification of 17 *Pseudomonas aeruginosa* srnas and prediction of srna-encoding genes in 10 diverse pathogens using the bioinformatic tool srnapredict2," *Nucleic Acids Res*, vol. 34, no. 12, pp. 3484–93, 2006.

[97] Livny, J., Fogel, M. A., Davis, B. M., and Waldor, M. K., "srnapredict: an integrative computational approach to identify srnas in bacterial genomes," *Nucleic Acids Res*, vol. 33, no. 13, pp. 4096–105, 2005.

[98] Lowe, T. M. and Eddy, S. R., "trnascan-se: a program for improved detection of transfer rna genes in genomic sequence," *Nucleic Acids Res*, vol. 25, no. 5, pp. 955–64, 1997.

[99] Lowe, T. M. and Eddy, S. R., "A computational screen for methylation guide snornas in yeast," *Science*, vol. 283, no. 5405, pp. 1168–71, 1999.

[100] Mao, X., Cai, T., Olyarchuk, J. G., and Wei, L., "Automated genome annotation and pathway identification using the kegg orthology (ko) as a controlled vocabulary," *Bioinformatics*, vol. 21, no. 19, pp. 3787–93, 2005.

[101] Matera, A. G., Terns, R. M., and Terns, M. P., "Non-coding rnas: lessons from the small nuclear and small nucleolar rnas," *Nat Rev Mol Cell Biol*, vol. 8, no. 3, pp. 209–20, 2007.

[102] Mattick, J. S., "Non-coding rnas: the architects of eukaryotic complexity," *EMBO Rep*, vol. 2, no. 11, pp. 986–91, 2001.

[103] Mattick, J. S. and Makunin, I. V., "Non-coding rna," *Hum Mol Genet*, vol. 15, no. Review issue 1, pp. R17–29, 2006.

[104] McCutcheon, J. P. and Eddy, S. R., "Computational identification of non-coding rnas in *Saccharomyces cerevisiae* by comparative genomics," *Nucleic Acids Res*, vol. 31, no. 14, pp. 4119–28, 2003.

[105] Meyer, I. M., "A practical guide to the art of rna gene prediction," *Brief Bioinform*, 2007.

[106] Mohanty, B. K. and Kushner, S. R., "Rho-independent transcription terminators inhibit rnase p processing of the secg leuu and mett trna polycistronic transcripts in *Escherichia coli*," *Nucleic Acids Res*, vol. 36, no. 2, pp. 364–75, 2008.

[107] Moreno-Hagelsieb, G. and Collado-Vides, J., "A powerful non-homology method for the prediction of operons in prokaryotes," *Bioinformatics*, vol. 18 Suppl 1, pp. S329–36, 2002.

[108] Ng, K. L. and Mishra, S. K., "De novo svm classification of precursor microrna from genomic pseudo hairpins using global and intrinsic folding measures," *Bioinformatics*, vol. 23, no. 11, pp. 1321–30, 2007.

[109] O'Hara, E. B., Chekanova, J. A., Ingle, C. A., Kushner, Z. R., Peters, E., and Kushner, S. R., "Polyadenylylation helps regulate mrna decay in *Escherichia coli*," *Proc Natl Acad Sci U S A*, vol. 92, no. 6, pp. 1807–11, 1995.

[110] Olman, V., Xu, D., and Xu, Y., "Cubic: identification of regulatory binding sites through data clustering," *J Bioinform Comput Biol*, vol. 1, no. 1, pp. 21–40, 2003.

[111] Omer, A. D., Lowe, T. M., Russell, A. G., Ebhardt, H., Eddy, S. R., and Dennis, P. P., "Homologs of small nucleolar rnas in archaea," *Science*, vol. 288, no. 5465, pp. 517–22, 2000.

[112] Osborne, R. J. and Thornton, C. A., "Rna-dominant diseases," *Hum Mol Genet*, vol. 15, no. Suppl 2, pp. R162–9, 2006.

[113] Ow, M. C. and Kushner, S. R., "Initiation of trna maturation by rnase e is essential for cell viability in *E. coli*," *Genes Dev*, vol. 16, no. 9, pp. 1102–15, 2002.

[114] Pang, K. C., Frith, M. C., and Mattick, J. S., "Rapid evolution of noncoding rnas: lack of conservation does not mean lack of function," *Trends Genet*, vol. 22, no. 1, pp. 1–5, 2006.

[115] Pedersen, J. S., Bejerano, G., Siepel, A., Rosenbloom, K., Lindblad-Toh, K., Lander, E. S., Kent, J., Miller, W., and Haussler, D., "Identification and classification of conserved rna secondary structures in the human genome," *PLoS Comput Biol*, vol. 2, no. 4, p. e33, 2006.

[116] Peng, Q., Wang, Z. J., and Liu, K. J. R., "Genomic processing for cancer classification and prediction: a broad review of the recent advances in model-based genomic and proteomic signal processing for cancer detection," *IEEE Signal Processing Magazine*, vol. 24, no. 1, pp. 100–10, 2007.

[117] Peterson, J. D., Umayam, L. A., Dickinson, T., Hickey, E. K., and White, O., "The comprehensive microbial resource," *Nucleic Acids Res*, vol. 29, no. 1, pp. 123–5, 2001.

[118] Pichon, C. and Felden, B., "Intergenic sequence inspector: searching and identifying bacterial rnas," *Bioinformatics*, vol. 19, no. 13, pp. 1707–9, 2003.

[119] Poole II, F. L., Gerwe, B. A., Hopkins, R. C., Schut, G. J., Weinberg, M. V., Jenney, F. E., J., and Adams, M. W., "Defining genes in the genome of

the hyperthermophilic archaeon *Pyrococcus furiosus* : implications for all microbial genomes," *J Bacteriol*, vol. 187, no. 21, pp. 7325–32, 2005.

[120] Price, M. N., Huang, K. H., Alm, E. J., and Arkin, A. P., "A novel method for accurate operon predictions in all sequenced prokaryotes," *Nucleic Acids Res*, vol. 33, no. 3, pp. 880–92, 2005.

[121] Rivas, E. and Eddy, S. R., "Secondary structure alone is generally not statistically significant for the detection of noncoding rnas," *Bioinformatics*, vol. 16, no. 7, pp. 583–605, 2000.

[122] Rivas, E. and Eddy, S. R., "Noncoding rna gene detection using comparative sequence analysis," *BMC Bioinformatics*, vol. 2, no. 1, p. 8, 2001.

[123] Rivas, E., Klein, R. J., Jones, T. A., and Eddy, S. R., "Computational identification of noncoding rnas in *E. coli* by comparative genomics," *Curr Biol*, vol. 11, no. 17, pp. 1369–73, 2001.

[124] Robb, F. T., Maeder, D. L., Brown, J. R., DiRuggiero, J., Stump, M. D., Yeh, R. K., Weiss, R. B., and Dunn, D. M., "Genomic sequence of hyperthermophile, *Pyrococcus furiosus* : implications for physiology and enzymology," *Methods in Enzymology*, vol. 330, pp. 134–57, 2001.

[125] Rogic, S., Mackworth, A. K., and Ouellette, F. B., "Evaluation of gene-finding programs on mammalian sequences," *Genome Res*, vol. 11, no. 5, pp. 817–32, 2001.

[126] Romero, P. R. and Karp, P. D., "Using functional and organizational information to improve genome-wide computational prediction of transcription units on pathway-genome databases," *Bioinformatics*, vol. 20, no. 5, pp. 709–17, 2004.

[127] Rudd, K. E., "Novel intergenic repeats of *Escherichia coli* k-12," *Res Microbiol*, vol. 150, no. 9-10, pp. 653–64, 1999.

[128] Sabatti, C., Rohlin, L., Oh, M. K., and Liao, J. C., "Co-expression pattern from dna microarray experiments as a tool for operon prediction," *Nucleic Acids Res*, vol. 30, no. 13, pp. 2886–93, 2002.

[129] Saetrom, P., Sneve, R., Kristiansen, K. I., Snove, O., J., Grunfeld, T., Rognes, T., and Seeberg, E., "Predicting non-coding rna genes in *Escherichia coli* with boosted genetic programming," *Nucleic Acids Res*, vol. 33, no. 10, pp. 3263–70, 2005.

[130] Salgado, H., Gama-Castro, S., Peralta-Gil, M., Diaz-Peredo, E., Sanchez-Solano, F., Santos-Zavaleta, A., Martinez-Flores, I., Jimenez-Jacinto, V., Bonavides-Martinez, C., Segura-Salazar, J., Martinez-Antonio, A., and Collado-Vides, J., "Regulondb (version 5.0): *Escherichia coli* k-12 transcriptional regulatory network, operon organization, and growth conditions," *Nucleic Acids Res*, vol. 34, no. Database issue, pp. D394–7, 2006.

[131] Salgado, H., Moreno-Hagelsieb, G., Smith, T. F., and Collado-Vides, J., "Operons in *Escherichia coli* : genomic analyses and predictions," *Proc Natl Acad Sci U S A*, vol. 97, no. 12, pp. 6652–7, 2000.

[132] Schattner, P., "Searching for rna genes using base-composition statistics," *Nucleic Acids Res*, vol. 30, no. 9, pp. 2076–82, 2002.

[133] Schattner, P., Barberan-Soler, S., and Lowe, T. M., "A computational screen for mammalian pseudouridylation guide h/aca rnas," *Rna*, vol. 12, no. 1, pp. 15–25, 2006.

[134] Schattner, P., Decatur, W. A., Davis, C. A., Ares, M., J., Fournier, M. J., and Lowe, T. M., "Genome-wide searching for pseudouridylation guide snornas: analysis of the *Saccharomyces cerevisiae* genome," *Nucleic Acids Res*, vol. 32, no. 14, pp. 4281–96, 2004.

[135] Schattner, P., *Computational Gene-finding for Non-coding RNAs.* Non-coding RNAs: Molecular Biology and Molecular Medicine, Georgetown, TX: Landes Bioscience, 2003.

[136] Schut, G. J., Brehm, S. D., Datta, S., and Adams, M. W., "Whole-genome dna microarray analysis of a hyperthermophile and an archaeon: *Pyrococcus furiosus* grown on carbohydrates or peptides," *J Bacteriol*, vol. 185, no. 13, pp. 3935–47, 2003.

[137] Shih, Y. L., Kawagishi, I., and Rothfield, L., "The mreb and min cytoskeletal-like systems play independent roles in prokaryotic polar differentiation," *Mol Microbiol*, vol. 58, no. 4, pp. 917–28, 2005.

[138] Shih-Chieh, S., Jay Kuo, C. C., and Ting, C., "Single nucleotide polymorphism data analysis: state-of-the-art review on this emerging field from a signal processing viewpoint," *IEEE Signal Processing Magazine*, vol. 24, no. 1, pp. 75–82, 2007.

[139] Sun-Yuan, K. and Man-Wai, M., "Machine learning for multimodality genomic signal processing," *IEEE Signal Processing Magazine*, vol. 23, no. 3, pp. 117–21, 2006.

[140] Szymanski, M., Barciszewska, M. Z., Zywicki, M., and Barciszewski, J., "Noncoding rna transcripts," *J Appl Genet*, vol. 44, no. 1, pp. 1–19, 2003.

[141] Tang, T. H., Bachellerie, J. P., Rozhdestvensky, T., Bortolin, M. L., Huber, H., Drungowski, M., Elge, T., Brosius, J., and Huttenhofer, A., "Identification of 86 candidates for small non-messenger rnas from the archaeon *Archaeoglobus fulgidus*," *Proc Natl Acad Sci U S A*, vol. 99, no. 11, pp. 7536–41, 2002.

[142] Tang, T. H., Polacek, N., Zywicki, M., Huber, H., Brugger, K., Garrett, R., Bachellerie, J. P., and Huttenhofer, A., "Identification of novel non-coding rnas as potential antisense regulators in the archaeon *Sulfolobus solfataricus*," *Mol Microbiol*, vol. 55, no. 2, pp. 469–81, 2005.

[143] Tatusov, R. L., Koonin, E. V., and Lipman, D. J., "A genomic perspective on protein families," *Science*, vol. 278, no. 5338, pp. 631–7, 1997.

[144] Tjaden, B., Goodwin, S. S., Opdyke, J. A., Guillier, M., Fu, D. X., Gottesman, S., and Storz, G., "Target prediction for small, noncoding rnas in bacteria," *Nucleic Acids Res*, vol. 34, no. 9, pp. 2791–802, 2006.

[145] Tran, T. T., Dam, P., Su, Z., Poole II, F. L., Adams, M. W., Zhou, G. T., and Xu, Y., "Operon prediction in *Pyrococcus furiosus*," *Nucleic Acids Res*, vol. 35, no. 1, pp. 11–20, 2007.

[146] Upadhyay, R., Bawankar, P., Malhotra, D., and Patankar, S., "A screen for conserved sequences with biased base composition identifies noncoding rnas in the a-t rich genome of *Plasmodium falciparum*," *Mol Biochem Parasitol*, vol. 144, no. 2, pp. 149–58, 2005.

[147] Uzilov, A. V., Keegan, J. M., and Mathews, D. H., "Detection of non-coding rnas on the basis of predicted secondary structure formation free energy change," *BMC Bioinformatics*, vol. 7, p. 173, 2006.

[148] Vaidyanathan, P. P., "Genomics and proteomics: a signal processor's tour," *IEEE Circuits and Systems Magazine*, vol. 4, no. 4, pp. 6–29, 2004.

[149] Venter, J. C., Adams, M. D., Myers, E. W., Li, P. W., Mural, R. J., Sutton, G. G., Smith, H. O., Yandell, M., Evans, C. A., Holt, R. A., Gocayne, J. D., Amanatides, P., Ballew, R. M., Huson, D. H., Wortman, J. R., Zhang, Q., Kodira, C. D., Zheng, X. H., Chen, L., Skupski, M., Subramanian, G., Thomas, P. D., Zhang, J., Gabor Miklos, G. L., Nelson, C., Broder, S., Clark, A. G., Nadeau, J., McKusick, V. A., Zinder, N., Levine, A. J., Roberts, R. J., Simon, M., Slayman, C., Hunkapiller, M., Bolanos, R., Delcher, A., Dew, I., Fasulo, D., Flanigan, M., Florea, L., Halpern, A., Hannenhalli, S., Kravitz, S., Levy, S., Mobarry, C., Reinert, K., Remington, K., Abu-Threideh, J., Beasley, E., Biddick, K., Bonazzi, V., Brandon, R., Cargill, M., Chandramouliswaran, I., Charlab, R., Chaturvedi, K., Deng, Z., Di Francesco, V., Dunn, P., Eilbeck, K., Evangelista, C., Gabrielian, A. E., Gan, W., Ge, W., Gong, F., Gu, Z., Guan, P., Heiman, T. J., Higgins, M. E., Ji, R. R., Ke, Z., Ketchum, K. A., Lai, Z., Lei, Y., Li, Z., Li, J., Liang, Y., Lin, X., Lu, F., Merkulov, G. V., Milshina, N., Moore, H. M., Naik, A. K., Narayan, V. A., Neelam, B., Nusskern, D., Rusch, D. B., Salzberg, S., Shao, W., Shue, B., Sun, J., Wang, Z., Wang, A., Wang, X., Wang, J., Wei, M., Wides, R., Xiao, C., Yan, C., and others, "The sequence of the human genome," *Science*, vol. 291, no. 5507, pp. 1304–51, 2001.

[150] Viegas, S. C., Pfeiffer, V., Sittka, A., Silva, I. J., Vogel, J., and Arraiano, C. M., "Characterization of the role of ribonucleases in salmonella small rna decay," *Nucleic Acids Res*, vol. 35, no. 22, pp. 7651–64, 2007.

[151] Voorhorst, W. G., Gueguen, Y., Geerling, A. C., Schut, G., Dahlke, I., Thomm, M., van der Oost, J., and de Vos, W. M., "Transcriptional regulation in the hyperthermophilic archaeon *Pyrococcus furiosus* : coordinated expression of divergently oriented genes in response to beta-linked glucose polymers," *J Bacteriol*, vol. 181, no. 12, pp. 3777–83, 1999.

[152] Wachi, M., Osaka, K., Kohama, T., Sasaki, K., Ohtsu, I., Iwai, N., Takada, A., and Nagai, K., "Transcriptional analysis of the *Escherichia coli* mrebcd genes responsible for morphogenesis and chromosome segregation," *Biosci Biotechnol Biochem*, vol. 70, no. 11, pp. 2712–9, 2006.

[153] Wang, C., Ding, C., Meraz, R. F., and Holbrook, S. R., "Psol: a positive sample only learning algorithm for finding non-coding rna genes," *Bioinformatics*, vol. 22, no. 21, pp. 2590–6, 2006.

[154] Wang, L., Trawick, J. D., Yamamoto, R., and Zamudio, C., "Genome-wide operon prediction in *Staphylococcus aureus*," *Nucleic Acids Res*, vol. 32, no. 12, pp. 3689–702, 2004.

[155] Wang, X., Zhang, J., Li, F., Gu, J., He, T., Zhang, X., and Li, Y., "Microrna identification based on sequence and structure alignment," *Bioinformatics*, vol. 21, no. 18, pp. 3610–4, 2005.

[156] Washietl, S., Hofacker, I. L., Lukasser, M., Huttenhofer, A., and Stadler, P. F., "Mapping of conserved rna secondary structures predicts thousands of functional noncoding rnas in the human genome," *Nat Biotechnol*, vol. 23, no. 11, pp. 1383–90, 2005.

[157] Washietl, S., Hofacker, I. L., and Stadler, P. F., "Fast and reliable prediction of noncoding rnas," *Proc Natl Acad Sci U S A*, vol. 102, no. 7, pp. 2454–9, 2005.

[158] Wassarman, K. M., Repoila, F., Rosenow, C., Storz, G., and Gottesman, S., "Identification of novel small rnas using comparative genomics and microarrays," *Genes Dev*, vol. 15, no. 13, pp. 1637–51, 2001.

[159] Weinberg, M. V., Schut, G. J., Brehm, S., Datta, S., and Adams, M. W., "Cold shock of a hyperthermophilic archaeon: *Pyrococcus furiosus* exhibits multiple responses to a suboptimal growth temperature with a key role for membrane-bound glycoproteins," *J Bacteriol*, vol. 187, no. 1, pp. 336–48, 2005.

[160] West, D., *Introduction to Graph Theory*. Upper Saddle River, NJ: Prentice Hall, Inc., 3rd ed., 2006.

[161] Westover, B. P., Buhler, J. D., Sonnenburg, J. L., and Gordon, J. I., "Operon prediction without a training set," *Bioinformatics*, vol. 21, no. 7, pp. 880–8, 2005.

[162] Workman, C. and Krogh, A., "No evidence that mrnas have lower folding free energies than random sequences with the same dinucleotide distribution," *Nucleic Acids Res*, vol. 27, no. 24, pp. 4816–22, 1999.

[163] Wu, H., Su, Z., Mao, F., Olman, V., and Xu, Y., "Prediction of functional modules based on comparative genome analysis and gene ontology application," *Nucleic Acids Res*, vol. 33, no. 9, pp. 2822–37, 2005.

[164] Xue, C., Li, F., He, T., Liu, G. P., Li, Y., and Zhang, X., "Classification of real and pseudo microrna precursors using local structure-sequence features and support vector machine," *BMC Bioinformatics*, vol. 6, p. 310, 2005.

[165] Yachie, N., Numata, K., Saito, R., Kanai, A., and Tomita, M., "Prediction of non-coding and antisense rna genes in *Escherichia coli* with gapped markov model," *Gene*, vol. 372, pp. 171–81, 2006.

[166] Yada, T., Nakao, M., Totoki, Y., and Nakai, K., "Modeling and predicting transcriptional units of *Escherichia coli* genes using hidden markov models," *Bioinformatics*, vol. 15, no. 12, pp. 987–93, 1999.

[167] Yan, B., Methe, B. A., Lovley, D. R., and Krushkal, J., "Computational prediction of conserved operons and phylogenetic footprinting of transcription regulatory elements in the metal-reducing bacterial family *Geobacteraceae*," *J Theor Biol*, vol. 230, no. 1, pp. 133–44, 2004.

[168] Yang, J. H., Zhang, X. C., Huang, Z. P., Zhou, H., Huang, M. B., Zhang, S., Chen, Y. Q., and Qu, L. H., "snoseeker: an advanced computational package for screening of guide and orphan snorna genes in the human genome," *Nucleic Acids Res*, vol. 34, no. 18, pp. 5112–23, 2006.

[169] Yao, Z., Weinberg, Z., and Ruzzo, W. L., "Cmfinder–a covariance model based rna motif finding algorithm," *Bioinformatics*, vol. 22, no. 4, pp. 445–52, 2006.

[170] Yoon, B.-J. and Vaidyanathan, P. P., "Computational identification and analysis of noncoding rnas: unearthing the buried treasure in the genome," *IEEE Signal Processing Magazine*, vol. 24, no. 1, pp. 64–74, 2007.

[171] Yousef, M., Nebozhyn, M., Shatkay, H., Kanterakis, S., Showe, L. C., and Showe, M. K., "Combining multi-species genomic data for microrna identification using a naive bayes classifier machine learning for identification of microrna genes," *Bioinformatics*, vol. 22, no. 11, pp. 1325–34, 2006.

[172] Zago, M. A., Dennis, P. P., and Omer, A. D., "The expanding world of small rnas in the hyperthermophilic archaeon *Sulfolobus solfataricus*," *Mol Microbiol*, vol. 55, no. 6, pp. 1812–28, 2005.

[173] Zhang, Y., Zhang, Z., Ling, L., Shi, B., and Chen, R., "Conservation analysis of small rna genes in *Escherichia coli*," *Bioinformatics*, vol. 20, no. 5, pp. 599–603, 2004.

[174] Zheng, Y., Szustakowski, J. D., Fortnow, L., Roberts, R. J., and Kasif, S., "Computational identification of operons in microbial genomes," *Genome Res*, vol. 12, no. 8, pp. 1221–30, 2002.

[175] Zuker, M., "Mfold web server for nucleic acid folding and hybridization prediction," *Nucleic Acids Res*, vol. 31, no. 13, pp. 3406–15, 2003.

# VITA

Thao Tran was born in Vietnam. She received her B.S. in Electrical Engineering, Cooperative Plan, Highest Honor from the Georgia Institute of Technology in May 2002. She received her M.S. in Electrical and Computer Engineering from the Georgia Institute of Technology in December 2003. Since then, she has been working towards her Ph.D. degree in Electrical and Computer Engineering at the Georgia Institute of Technology. From 1999-2001, she worked at Motorola's Platform Software Division Group and Hardware Development Group testing various software and hardware issues for a PDA/phone device.

She was awarded the National Science Foundation Graduate Research Fellowship and the President's Fellowship from the Georgia Institute of Technology in 2002. Thao Tran's research interests are in the areas of genomic signal processing, pattern recognition, machine learning, and bioinformatics.