Improving Algorithms of Gene Prediction in Prokaryotic Genomes, Metagenomes, and Eukaryotic Transcriptomes

A Dissertation
Presented to
The Academic Faculty

by

Shiyuyun Tang

In Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy in Bioinformatics in the
School of Biology

Georgia Institute of Technology
May 2016

Improving Algorithms of Gene Prediction in Prokaryotic Genomes, Metagenomes, and Eukaryotic Transcriptomes

Approved by:

Dr. Mark Borodovsky, Advisor
Department of Biomedical Engineering and Computational Science and Engineering
Georgia Institute of Technology

Dr. Kostas T. Konstantinidis
School of Civil and Environmental Engineering
Georgia Institute of Technology

Dr. Brian Hammer
School of Biology
Georgia Institute of Technology

Dr. Greg Gibson
School of Biology
Georgia Institute of Technology

Dr. Srinivas Aluru
School of Computational Science and Engineering
Georgia Institute of Technology

Date Approved: March 29, 2016

For my family and all the great teachers in my life

ACKNOWLEDGEMENTS

I would like to extend my deepest gratitude to my advisor Dr. Mark Borodovsky for his invaluable guidance. My success would not have been possible without the continuous support and nurturing of him. I also would like to extend my sincere thanks to Dr. Alex Lomsadze who has offered a tremendous amount of knowledge, assistance, and encouragement throughout the duration of my PhD. I am also grateful to my committee members, Dr. Kostas Konstantinidis, Dr. Brian Hammer, Dr. Greg Gibson, and Dr. Srinivas Aluru for their insightful questions and discussions on my thesis proposal and for reviewing this dissertation. I also wish to thank my family, my parents Yin Yu and Lun Tang and my boyfriend Ernesto A. Estrada for their unconditional support and love.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF SYMBOLS AND ABBREVIATIONS

BLAST	basic local alignment search tool
cDNA	complementary DNA, formed by reverse transcription of mRNA
CDS	coding sequence
COGs	clusters of orthologous groups of proteins
EST	expressed sequence tag
FN	false negative
FP	false positive
FS	frameshift
GC content	percentage of G and C nucleotides in a sequence
HMM	hidden Markov model
HSMM	hidden semi-Markov model
ICM	interpolated context model
IMM	interpolated Markov model
NCBI	National Center for Biotechnology Information
NGS	next-generation sequencing
ORF	open reading frame
PAI	pathogenicity island
PAIDB	pathogenicity island database
PFM	positional frequency matrix
PGAP	prokaryotic genome annotation pipeline
PNNL	the Pacific Northwest National Laboratory
ps ORFs	peptide-supported open reading frames

RBS	ribosomal binding site
RGASP	RNA-Seq Genome Annotation Assessment Project
Sn	sensitivity: TP/(TP+FN)
Sp	specificity: TP/(TP+FP)
SVM	support vector machine
TIS	translation initiation site
TP	true positive
UTRs	untranslated regions

SUMMARY

The next-generation sequencing technology has generated enormous amount of DNA and RNA sequences that potentially contain volumes of important genetic information, *e.g.* information on protein-coding genes. The goal of research described in this thesis was to improve prediction of protein-coding genes in newly sequenced genomes by the algorithms and software tools of the GeneMark line. The thesis is divided into three main parts describing i) GeneMarkS-2, ii) GeneMarkS-T, and iii) MetaGeneTack.

In prokaryotic genomes, *ab initio* gene finders can predict genes with high accuracy. However, the error rate is not negligible and largely species-specific. Most errors in gene prediction are made in genes located in genomic regions with atypical GC composition, e.g. genes in pathogenicity islands. We describe a new algorithm GeneMarkS-2 that uses local GC-specific heuristic models for scoring individual ORFs in the first step of analysis. Predicted atypical genes are retained and serve as 'external' evidence in subsequent runs of self-training. GeneMarkS-2 also controls the quality of training process by effectively selecting optimal orders of the Markov chain models as well as duration parameters in the hidden semi-Markov model. GeneMarkS-2 has shown significantly improved accuracy compared with other state-of-the-art gene prediction tools.

Massive parallel sequencing of RNA transcripts by the next generation technology (RNA-Seq) provides large amount of RNA reads that can be assembled to full transcriptome. We have developed a new tool, GeneMarkS-T, for *ab initio* identification

of protein-coding regions in RNA transcripts. Unsupervised estimation of parameters of the algorithm makes unnecessary several steps in the conventional gene prediction protocols, most importantly the manually curated preparation of training sets. We have demonstrated that the GeneMarkS-T self-training is robust with respect to the presence of errors in assembled transcripts and the accuracy of GeneMarkS-T in identifying protein-coding regions and, particularly, in predicting gene starts compares favorably to other existing methods.

Frameshift prediction (FS) is important for analysis and biological interpretation of metagenomic sequences. Reads in metagenomic samples are prone to sequencing errors. Insertion and deletion errors that change the coding frame impair the accurate identification of protein coding genes. Accurate frameshift prediction requires sufficient amount of data to estimate parameters of species-specific statistical models of protein-coding and non-coding regions. However, this data is not available; all we have is metagenomic sequences of unknown origin. The challenge of *ab initio* FS detection is, therefore, twofold: (i) to find a way to infer necessary model parameters and (ii) to identify positions of frameshifts (if any). We describe a new tool, MetaGeneTack, which uses a heuristic method to estimate parameters of sequence models used in the FS detection algorithm. It was shown on several test sets that the performance of MetaGeneTack FS detection is comparable or better than the one of earlier developed program FragGeneScan.

The work presented in this dissertation contributed to the following publications:

Tang, Shiyuyun, Alexandre Lomsadze, Karl Gemayel, and Mark Borodovsky. "Improving *ab initio* Gene Prediction in Prokaryotic Genomes" *Nucleic acids research* (submitted).

Tang, Shiyuyun, Alexandre Lomsadze, and Mark Borodovsky. "Identification of protein coding regions in RNA transcripts." *Nucleic acids research* (2015): gkv227.

Tang, Shiyuyun, Ivan Antonov, and Mark Borodovsky. "MetaGeneTack: *ab initio* detection of frameshifts in metagenomic sequences." *Bioinformatics*29.1 (2013): 114-116.

CHAPTER 1
INTRODUCTION

The ever accelerating accumulation of DNA and RNA sequences is due to revolutionary changes in sequencing technology. As depicted in Figure 1.1, the number of sequenced genomes in the National Center for Biotechnology Information (NCBI) genome database is growing exponentially over the last 20 years. These data demand highly automated tools for accurate genome annotation. As a key component of genome annotation, gene finding aims at locating the endpoints (the start and stop) of all protein coding genes for which two major approaches have been developed: homology-based methods and *ab initio* methods. One of the focuses of this work is describing a new *ab initio* gene finder GeneMarkS-2, developed upon a line of GeneMark tools, which aims at closing the open endings of gene prediction in prokaryotic genomes.

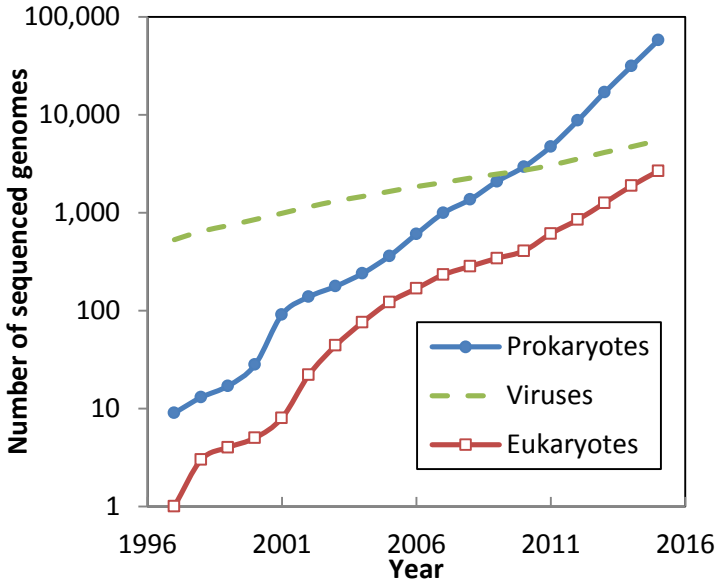

Figure 1.1 The growth of sequenced genomes curated by NCBI

Massive parallel sequencing of RNA transcripts by next-generation technology (RNA-Seq) (Wang, et al., 2009) also generates critically important data for gene discovery. Many computational tools are able to reconstruct the full-length representation of cellular RNAs from the vast amount of RNA-Seq reads (Grabherr, et al., 2011; Guttman, et al., 2010; Haas, et al., 2013; Li, et al., 2011; Mezlini, et al., 2013; Roberts, et al., 2011; Robertson, et al., 2010; Schulz, et al., 2012; Slater and Birney, 2005). The accuracy of transcript reconstruction by a large array of assembly tools is evaluated by the international RNA-Seq Genome Annotation Assessment Project (RGASP) consortium (Steijger, et al., 2013).

Eukaryotic transcripts in the spliced form share the same characteristics as prokaryotic genomic sequences: both sequences contain intron-less genes. Therefore the development of gene-finding algorithms for prokaryotic genomes has laid a solid foundation for gene finding of transcriptomic data. Similarly, in the emerging field of meta-transcriptomics (microbiome-wide gene expression profiling through RNA-Seq) short meta-transcriptomic reads can be assembled into transcripts encoding one or more genes, which provide valuable information for gene prediction. It has been shown that reconstruction of meta-transcriptome datasets significantly improves the functional annotation of sequence reads (Sekhar, et al., 2011).

Although a steady progress has been made on transcript reconstruction, very few tools are dedicated to identifying the protein coding-regions in the sequences, with little assessment of the overall performance on error-prone transcripts. In this work, we present a self-training algorithm called GeneMarkS-T (‘S’ stands for Self-training and ‘T’ stands for Transcripts) that is suitable for *ab intio* gene prediction in transcripts.

The last part of the work is on frameshift prediction in metagenomic sequences. A frameshifts is caused by insertion or deletion of nucleotides in a coding sequence. The number of inserted or deleted nucleotides is not divisible by three, causing a shift in the reading frame (the grouping of the codons). Frameshifts could be a result of sequencing error, indel (insertion or deletion) mutation inside protein-coding regions, or a recoding event (Baranov, et al., 2005; Decatur and Fournier, 2003; Maas, 2012; Sharma, et al., 2011; Wernegreen, et al., 2010). Frameshifts greatly affect gene prediction as they disrupt the open reading frame (ORF) of the gene and subsequently change the protein product from the ORF.

Sequencing-error-induced frameshifts are more prominent in metagenomic sequences. Since the short reads of sequences come from a microbial community, they are less homogeneous, more difficult to assemble, and more difficult for error correcting than the genomic sequences. Error rates in metagenomic sequences depend on various factors including species complexity in the metagenomic sample, genome abundance, the sequencing method, and assembly strategies (Luo, Tsementzi, Kyrpides and Konstantinidis, 2012). Therefore it is desirable to predict and correct frameshift error in metagenomic sequences before gene annotation. We describe a tool 'MetaGeneTack' that can effectively predict frameshifts in metagenomic sequences.

1.1 Gene finding in prokaryotic genomes

Similarity-based methods and *ab initio* methods are two main streams of approaches for gene prediction in prokaryotic genomes. In similarity-based methods the sequence of interest, a DNA sequence or its translated version, is searched against databases of known genes using BLAST-type (Altschul, et al., 1997) mapping techniques. If a homolog with significant similarity is found, the sequence of interest is characterized as a gene. This method can give high sensitivity and specificity if close relatives exist in the database (to reach high sensitivity) and no errors are present in the

database (to reach high specificity). However, the method would fail to identify a novel gene with no homologs in the database. If the database contain hypothetical or uncharacterized genes that are in nature non-coding sequences, such errors would propagate during annotation of a new gene. In addition, similarity-based approach cannot predict gene start and short genes precisely.

Unlike similarity-based methods, a*b initio* gene prediction approaches, also referred to as "intrinsic methods", do not depend on the existence or the quality of gene databases. These methods use intrinsic features of the given sequence for gene identification. Statistical models such as Markov models, hidden Markov models (HMM), and hidden semi-Markov models (HSMM, also called "hidden Markov model with duration" or "generalized hidden Markov model (GHMM)") proved to be very useful for modeling statistical patterns of nucleotide ordering in protein-coding and non-coding regions.

This section reviews the widely used gene prediction tools and algorithms, with a focus on *ab initio* approaches. We only discuss gene finding in prokaryotic sequences, as eukaryotic sequences contain introns and the gene prediction in eukaryotes is out of the scope of this thesis.

1.1.1 The GeneMark-line of gene finders

A serial of *ab initio* gene-finders and related algorithms for training model parameters required for the gene finders have been developed in the group lead by *Mark Borodovsky*. All of the algorithms described here can be accessed through the website http://exon.gatech.edu/GeneMark/. The GeneMark-line of algorithms forms the basis of the development of GeneMarkS-2, GeneMarkS-T, and MetaGeneTack.

1.1.1.1 GeneMark

Before the publication of GeneMark, a pioneer work in 1986 (Borodovsky, et al., 1986) has shown that nucleotide frequencies are different in coding and non-coding DNA. This work analyzed fragments of coding and non-coding sequences of *Escherichia coli* and introduced three-periodic (frame-dependent) Markov chain models for characterizing coding DNA. Based on those findings, in 1993 GeneMark was introduced (Borodovsky and Mcininch, 1993). It employed the nonhomogeneous three-periodic Markov model and the Bayes' formalism for classification of sequences. GeneMark was able to recognize genes in both strands (all six frames) simultaneously, which was the first method of this kind.

In GeneMark, the three-periodic first-order (or higher order) Markov chain model of a coding region is defined by the three vectors of initial probabilities $\mathbf{P1_0, P2_0, P3_0}$ and the three transition matrices **P1, P2, P3** with the numbers 1, 2, 3 corresponding to the three codon positions. Each initial state vector contains the probabilities of A, C, G, T. Each transition matrix contains the probabilities of X given Y, with $X, Y \in \{A, C, G, T\}$. The parameters are derived from a set of training sequences of coding regions based on the maximum likelihood principle. Particularly they are frequencies calculated from the counts of mono- and di-nucleotides in each codon position of the training sequences. Similarly, the parameters for the non-homogeneous Markov chain model of the shadow of the coding region can be designated as $\mathbf{P4_0, P5_0, P6_0}$ and **P4, P5, P6**. The first-order Markov chain model of a non-coding region is homogeneous; it is also defined by a vector of initial probabilities and a transition matrix. Parameters are trained from known non-coding sequences.

The probability of a stretch of nucleotides "f_1, f_2… f_n" denoted as *F* observed in a given model can be calculated. For example, the probability of *F* appearing in a non-coding region can be calculated from the equation 1. The probability of *F* appearing in a coding region in frame one (with the first nucleotide appearing in the first codon position) can be calculated using equation 2. Finally, a set of *a posteriori* probabilities are defined using Bayes' theorem (equation 3 and 4). *P(NON)* and $P(COD_i)$ stand for the *a priori* probability of the event and *P(NON)=0.5* and $P(COD_i)=1/12$.

$$P(F \mid NON) = PN_0(f_1) * PN(f_2 \mid f_1) * ... * PN(f_n \mid f_{n-1}) \quad \textbf{(1)}$$

$$P(F \mid COD_1) = P1_0(f_1) * P1(f_2 \mid f_1) * P2(f_3 \mid f_2) * P3(f_4 \mid f_3) ... * P2(f_n \mid f_{n-1}) \quad \textbf{(2)}$$

$$P(COD_i \mid F) = \frac{P(F \mid COD_i) * P(COD_i)}{\sum_j P(F \mid COD_j) * P(COD_j) + P(F \mid NON) * P(NON)} \quad \textbf{(3)}$$

$$P(NON \mid F) = \frac{P(F \mid NON) * P(NON)}{\sum_j P(F \mid COD_j) * P(COD_j) + P(F \mid NON) * P(NON)} \quad \textbf{(4)}$$

The GeneMark web interface implements a graphical output of the posterior probability in each coding frame along a sequence (Besemer and Borodovsky, 2005) (see Figure 2.11 as an example of a GeneMark graph). The graph shows six panels, each representing a coding frame. In each panel, the vertical axis denotes the value of the *a posteriori* probability of a sequence being in the coding frame. The horizontal axis represents nucleotide positions along the sequence. The *a posteriori* probability of each position is calculated for the sequence fragment in a sliding window with this position

situated in the middle. Open reading frames are marked by solid lines. The GeneMark graph is a useful tool for visualizing coding potential along a sequence.

GeneMark was a pioneering gene finder used to annotate the first completely sequenced genomes (Blattner, et al., 1997; Bult, et al., 1996; Fraser, et al., 1995; Himmelreich, et al., 1996; Klenk, et al., 1997; Kunst, et al., 1997; Smith, et al., 1995).

1.1.1.2 GeneMark.hmm

The GeneMark program assumes the stretch of a given sequence to be complete coding or non-coding. GeneMark is able to identify the open reading frame where a gene resides, but it poses uncertainty of the 5' boundary of the gene. GeneMark.hmm (Lukashin and Borodovsky, 1998) was designed to solve this problem by incorporating the GeneMark approach into a hidden Markov model (HMM) framework.

In an HMM, for an observed DNA sequence $S = \{s_1, s_2, ..., s_L\}$ where s_i stands for a nucleotide *A*, *C*, *G*, or *T* and *L* is the length of the sequence, we can define a sequence of hidden states of the nucleotides as $A = \{a_1, a_2, ..., a_L\}$. GeneMark.hmm describes 9 hidden states: 1) the non-coding state, 2) the direct start codon, 3) the direct stop codon, 4) the direct typical coding state, 5) the direct atypical coding state, 6) the reverse start codon, 7) the reverse stop codon, 8) the reverse typical coding state, and 9) the reverse atypical coding state. Each $a_i \in \{1, 2, ..., 9\}$ denotes the hidden state that emits the nucleotide s_i. In GeneMark.hmm the coding and non-coding states are allowed to generate a stretch of nucleotides instead of one. The length of the stretch of sequence is called 'duration'. Under this framework, the hidden states of the sequence *S* can be

represented as $A=\{(a_1d_1),(a_2d_2),...,(a_md_m)\}$, in which d_i is the duration of a_i and $\sum d_i = L$. The optimal trajectory of the hidden states A^* is defined as the trajectory of A that gives the maximal conditional probability $P(A|S)$. A^* can be found using the standard Viterbi algorithm (Rabiner, 1989).

Parameters of the HMM are derived from annotated *E. coli* sequences. As described in the GeneMark paper, the three-periodic inhomogeneous Markov chain model was built from the known gene sequences in *E. coli*. The Markov chain model for the non-coding states was calculated from known non-coding sequences. The start codon probabilities equal the frequencies of ATG, GTG, and TTG observed in the genome. The duration parameters are calculated from the analytical frequency distribution of the lengths of coding and non-coding regions in *E. coli*. Genes were cluster into typical and atypical genes (Hayes and Borodovsky, 1998). Atypical genes refer to genes horizontally transferred into the genome from foreign sequences.

As the framework of HMM in GeneMark.hmm prevents the prediction of overlapping genes, a post-processing step was added to refine start prediction. The post-processing algorithm searches the -19 to -4 nt upstream sequences of the start of each predicted gene for a putative ribosomal binding site (RBS). The RBS model was in the form of a positional nucleotide frequency matrix (PFM), denoting the probability of observing A, C, G, or T in each position of the 5nt motif. Parameters of the PFM are derived through multiple sequence alignment of the upstream sequences of *E.coli* genes (Lukashin, et al., 1992). The final output of GeneMark.hmm shifts the predicted gene

start if the probability of the RBS in an alternative start upstream of the predicted one is larger than some threshold.

1.1.1.3 GeneMarkS

Developed in 2001, GeneMarkS (Besemer, et al., 2001) is a self-training algorithm that runs the GeneMark.hmm program iteratively to build model parameters for the HMM and finds the maximum likelihood parse of the hidden states of a given sequence. GeneMarkS achieved two major improvements. First, it derives genome-specific model parameters required for gene prediction in an unsupervised fashion, which can be applied on anonymous or novel genomic sequences without known proteins or any external training data. This improvement is a very important innovation for *ab initio* gene prediction, because as an increasing number of new species are being sequenced each year, waiting for curated training sequences would make an *ab initio* gene finder much less practical. Second, a new program GeneMark.hmm 2.0 is implemented in GeneMarkS. This new version integrates the RBS model into the HMM framework instead of using it as a post-processing step, which improves the accuracy of predicting gene start. In addition, GeneMark.hmm 2.0 allows the prediction of overlapping genes. As a result, the full length of a sequence can be parsed into coding and non-coding regions in one run without further adjustment.

The self-training algorithm works as follows. GeneMark.hmm 2.0 starts the first run of gene prediction with a set of heuristic model parameters (see 1.1.1.4). The initial set of predicted genes serve as a training set to build and update the coding, non-coding, start codon, and RBS model. Then the new set of model parameters is used to predict genes again on the sequence. The program runs iteratively between the prediction step

and training step until convergence, which means the change of prediction from two subsequent iterations is less than a small value. Predicted genes along with the model parameters in the last iteration are delivered as output. There is another option to add atypical genes predicted by the heuristic model to the final prediction.

A two-component RBS model is part of the training process. The model includes a position frequency matrix describing the RBS motif, and a spacer distribution describing the length between the start codon and the RBS motif. In each iteration, the upstream sequences of predicted genes are collected, and multiple sequence alignment is performed by a Gibbs sampling procedure (Lawrence, et al., 1993; Neuwald, et al., 1995) to find a conserved motif without gaps. The distance between the motifs found in the upstream sequences and the predicted gene starts is used to build the spacer length distribution.

A special version of GeneMarkS called GeneMarkS-plus has served as the core element of the National Center for Biotechnology Information prokaryotic genome annotation pipeline (PGAP)[1]; in August 2015 PGAP annotated and re-annotated more than 48,000 prokaryotic genomes. GeneMarkS-plus can incorporate external protein evidence into the *ab initio* prediction.

Chapter 2 discusses the new development of GeneMarkS called GeneMarkS-2. Chapter 3 discusses the modified GeneMarkS training algorithm that can be applied to gene prediction in RNA transcripts.

---

[1] http://www.ncbi.nlm.nih.gov/genome/annotation_prok/process/

1.1.1.4 Heuristic models and MetaGeneMark

The heuristic approach to derive model parameters for *ab initio* gene prediction was first introduced in 1999 (Besemer and Borodovsky, 1999). As discussed above, the HMMs employed in gene finders require accurate model parameters, especially parameters for the high-order non-homogeneous coding model. These parameters are species or genome specific. They can be derived from experimentally validated training sequences or a large enough set of anonymous sequences of the same species/genome. For a short sequence with unknown origin, neither of these training sequences exists. The proposed heuristic approach solves this problem innovatively by leveraging the relationship between nucleotide frequencies in the three codon positions and the global nucleotide frequencies as well as relationship between the amino acid frequencies and the genome GC content. Interestingly, the heuristic approach turns out to be extremely useful in the field of metagenomics that had not emerged until several years later.

Here I summarize the procedure of building the heuristic model described in (Besemer and Borodovsky, 1999). The first 17 genomes and their annotated genes available back then were used to build two linear relationships. The first one is between the global nucleotide frequency in the genome and the nucleotide frequency in each of the codon position in genes. For example, the global frequency of Thymine (T) and the frequency of T in the first position of all codons have the following linear relationship in the 17 genomes: $f(T)_{pos1} = 0.185 + 0.521(f(T)_{global} - 0.228)$. The second relationship is between the genome GC content and the amino acid frequency in the genome. For example, the frequency of amino acid alanine and the genome GC% has the following relationship: $f_{alanine}(GC\%) = 0.0749 + 0.0019 * (GC\% - 42.53)$.

From a given genomic sequence with certain nucleotide composition, the nucleotide frequency in each codon position is determined from the first linear relationship. Note that the global nucleotide composition can be represented by the global GC content as stated by the second Chargaff rule. The initial values of frequency of 61 codons $f_1(XYZ)$ are calculated as a product of frequency of the three nucleotides in corresponding codon positions. For example: $f_1(GCT) = f(G)_{pos1} * f(C)_{pos2} * f(T)_{pso3}$. The calculated initial codon frequencies are refined using amino acid frequencies determined by global GC%. As an example, the refined frequency of codon GCT that encodes alanine is shown in equation 5. Once all codon frequencies are determined, it is straightforward to calculate the parameters of zero-order three-periodic Markov model. For higher order Markov models, the transition probability matrix is built using di-codons assuming that the occurrence of adjacent codons is independent.

$$f_R(GCT) = f_{alanine}(GC\%) * \frac{f_1(GCT)}{f_1(GCA) + f_1(GCC) + f_1(GCG) + f_1(GCT)} \quad \textbf{(5)}$$

The heuristic model parameters have since been used for gene finding in short sequences such as genomes of viruses and plasmids and for initializing self-training as described in section 1.1.1.3. In the early to middle 2000's, with the advent of shotgun metagenomics and then high-throughput sequencing, a new application of heuristic model has emerged in the field of metagenomics. A metagenomic sample consists of a mixture of genetic materials from microbial communities taken from the environment. The majority of species in metagenomic samples has never been documented and cannot be cultivated in a laboratory. Therefore, many proteins encoded from genes in a new

metagenomic sample have no known homologs in existing protein databases. *Ab initio* gene prediction enabled by the heuristic model parameters becomes essential for identifying those new genes.

In 2010, a new set of heuristic models was introduced with the application for gene prediction in metagenomic sequences (Zhu, et al., 2010). Several improvements were made in the new heuristic approach. First, the number of training genomes was largely expanded. Compared with only 17 genomes used in the 1999 approach, 582 complete genomes and their annotated genes were used to build the new heuristic parameters. Second, since much more genomes including a set of archaea were available, *Zhu, et al.* was able to build two sets of heuristic models, one for bacteria and one for archaea. Similarly, they also divided genomes to be mesophilic or thermophilic and built two corresponding models. Third, they described and compared several new ways of predicting codon frequencies from the genome GC content. One of the new methods that produced good accuracy in gene prediction was through direct third-order polynomial regression of codon frequencies over genome GC content.

The 2010 paper also introduced MetaGeneMark, a program that combines the new heuristic model with the GeneMark.hmm algorithm to predict genes in metagenomic sequences. MetaGeneMark gives high (>90%) sensitivity and specificity in finding genes in short sequences (of several hundred nucleotides in length) of unknown origin and has since been used in many metagenomic projects (Forsberg, et al., 2014; Karlsson, et al., 2013; Nielsen, et al., 2014; Tyakht, et al., 2013).

1.1.2 Other *ab initio* gene-finders for prokaryotic genomes

Besides the GeneMark-line of gene finders, another popular *ab initio* gene finder is Glimmer (Delcher, et al., 2007; Delcher, et al., 1999; Dyer, et al., 2011; Kelley, et al., 2012). It uses interpolated Markov models (IMM), a linear combination of probabilities of various orders, giving higher weights to oligomers with more sufficient content. There are three generations of Glimmer. Glimmer1 (Salzberg, et al., 1998) built IMMs from annotated training sequences to score individual ORFs in all six frames. ORFs that score higher than a threshold in the correct frame are then resolved for overlaps. Glimmer2 (Delcher, et al., 1999) introduced an improved IMM called the interpolated context model (ICM), which can build dependencies from nucleotides not immediately adjacent to each other. As Glimmer1 produced many false negatives because of long overlaps and wrong start prediction, Glimmer2 also introduced other rules to resolve overlap such as an RBSfinder for post-processing gene start locations. In Glimmer3 (Delcher, et al., 2007), the RBS model is presented in the form of a position weight matrix, and was built by ELPH[2], a Gibbs sampling approach from multiple alignment of upstream regions of predicted genes. A big improvement of Glimmer3 compared with previous versions of Glimmer is that for the first time it integrates all gene predictions across an entire genome. All start positions are scored using the IMM and the RBS model and a global dynamic programming algorithm is used to select ORF starts that give maximum total score along the whole genome. Short overlaps are allowed as the dynamic programming

[2] http://www.cbcb.umd.edu/software/ELPH/

algorithm can backtrack within the maximum allowed overlap distance and update the total score of the path.

Accurate *ab initio* gene finding in isolated genomes requires ample sequence data for the estimation of algorithm parameters (model training). Glimmer1 uses very long ORFs of the given genome as a training set for coding model parameters. Later versions of Glimmer improve training by more cautiously selecting and filtering the training set. For example, Glimmer3 filter the initial set of long ORFs based on amino-acid composition.

EasyGene (Larsen and Krogh, 2003; Nielsen and Krogh, 2005) is another HMM-based gene finder. It also estimates the statistical significance of a predicted gene. To find training sequences, EasyGene translates all ORFs longer than a threshold in the given genomic sequence, and uses BLASTP (Altschul, et al., 1990) to search against the Swiss-Prot (Boeckmann, et al., 2003) database. This approach is dependent on the existence and the quality of protein databases, and the protein search may also increase run time.

A more recently developed *ab initio* gene finder, Prodigal (Hyatt, et al., 2010), does not follow a standard Markov model framework. It scores individual ORFs using various features and scoring rules and then performs dynamic programming on all pairs of start-and-stop triplets to find the maximum scoring path. Adopted features in Prodigal include GC bias in the first, second, and third positions of each codon, the frequency of hexamers, the ORF length, upstream letters resembling the RBS, etc. The training rules and parameters are fine-tuned on a set of curated genomes and are validated with a larger set of over 100 genomes from the GenBank annotation.

Another tool called GISMO (Gene Identification using Support vector Machine for ORF classification) is based on a support vector machine (SVM) with a Gaussian kernel to classify ORFs (Krause, et al., 2007). The SVM's features are 64-dimensional vectors of relative codon frequencies, as they yield the best classification performance. To train the SVM, GISMO searched all ORFs in the given genome against the PFAM protein database (Bateman, et al., 2004). The set of ORFs with good protein domain match forms the positive training set. ORFs overlapping the positive set comprise the negative training set. Note that all ORFs supported by strong protein domain evidence (e-value <0.1) are kept in the final prediction. ORFs with high SVM score but weak evidence are subject to removal as a result of long (>50bp) overlaps. Similar to EasyGene, GISMO also relies on the existence and quality of protein databases, which makes it a mixture of similarity-based and intrinsic approach rather than a pure *ab initio* gene finder.

1.1.3 Gene finding based on external evidence

Two examples of gene finders based on external evidence are ORPHEUS (Frishman, et al., 1998) and CRITICA (Badger and Olsen, 1999). ORPHEUS is one of the earliest extrinsic tools. It uses DPS program (Huang, 1996) to map known proteins to the query genomic sequence. Regions with high-scoring match are considered the "seed ORFs". Codon frequencies of the seed ORFs are used to compute coding quality parameters to evaluate the coding quality of other putative genes in the sequence. A putative gene is accepted if it is longer than 300bp and its coding quality is sufficiently high. ORPHEUS also uses an RBS model to refine start prediction, which is derived from the upstream sequences of seed ORFs with no alternative candidate starts.

Different from ORPHEUS, CRITICA uses BLASTN (Altschul, et al., 1990) to detect conserved stretches on the DNA level. Then it computes the di-codon statistics from the aligned high score regions. It also computes a score for the start site based on the quality of match to the Shine-Dalgarno sequence (the RBS) (Shine and Dalgarno, 1974). CRITICA shows high sensitivity but high false negative rate.

1.2 Gene prediction in EST sequences and transcripts

An mRNA transcript mirrors the DNA sequence from which it is transcribed. Studying of mRNAs provides valuable information about protein-coding genes in two aspects. Fist, constructing the full length of (coding) mRNAs and then looking for the coding ORF in the sequences is the most direct and reliable way for identifying protein-coding genes, especially in eukaryotes. Unlike genes in prokaryotes, genes in eukaryotes contain introns spliced out from the mature mRNA and thus excluded from translation. As a result, looking for the complete intron-exon structure for a final protein product on the DNA level is a challenging task. Adding to the complexity is the prevalence of alternative splicing, in which a single gene codes for multiple proteins through altering the intron-exon structure. Second, studying the full collection of mRNA transcripts can also tell us when and where each gene is turned on or off – the pattern of gene expression – which is essential in understanding disease and development.

An ideal method to obtain transcript sequences is to sequence the full mRNA molecule, or in practice sequence the full cDNA molecule created from mRNA through reverse transcription. However, this approach is extremely laborious and expensive, resulting in very limited coverage of all genes even for model organisms to date. Another method is through expressed sequence tags (ESTs). ESTs are short reads (200-800bp in

length (Nagaraj, et al., 2007)) derived from cDNA libraries (typically from the 5' and 3' ends of the cDNA molecule). ESTs can be generated through Sanger sequencing and later through next generation sequencing (NGS) with relatively low cost. This process was proposed and used to identify genes in the human genome before the full genome assembly was available. (Adams, et al., 1991; Boguski, 1995).

Unlike the low-throughput single-read sequencing of cDNA and ESTs, high-throughput RNA sequencing (RNA-seq) has promised tremendous opportunities towards mapping the comprehensive transcriptome (Wang, et al., 2009). Short RNA-Seq reads are obtained from fragmented mRNAs or cDNA and sequenced using NGS (high-throughput parallel sequencing) technologies. Reads generated by RNA-Seq can cover the full transcriptome with deep coverage and low cost. These advantages have made it a transformational tool for guiding gene prediction (Hoff, et al., 2015; Reid, et al., 2014), discovering novel transcripts (Roberts, et al., 2011; Trapnell, et al., 2010), and characterizing gene expression (Mortazavi, et al., 2008; Trapnell, et al., 2013; Trapnell, et al., 2012).

As RNA-Seq data become ubiquitous, many assembly algorithms have been developed to assemble RNA-Seq reads into longer transcripts (Grabherr, et al., 2011; Guttman, et al., 2010; Haas, et al., 2013; Li, et al., 2011; Mezlini, et al., 2013; Roberts, et al., 2011; Robertson, et al., 2010; Schulz, et al., 2012; Slater and Birney, 2005). Some studies have also used RNA-Seq to improve gene prediction and splice-junction characterization in genomes (Hoff, et al., 2015; Lomsadze, et al., 2014; Trapnell, et al., 2009; Wang, et al., 2010). However, there are few tools that allow direct gene calling on assembled transcripts.

A straightforward strategy of gene finding in transcript is to map translated transcripts to known proteins. Several such tools were developed earlier for EST and cDNA sequences. For example, OrfPredictor (Min, et al., 2005) uses BLASTX to map six-frame translation of the EST to protein databases. For ESTs with no significant hit, intrinsic features are used to predict coding regions. Similar to the case of prokaryotic gene finding, the alignment-based strategy will be successful only if the protein products have known homologs in protein databases.

ESTScan (Iseli, et al., 1999) is an HMM-based *ab initio* gene finder designed for low-quality ESTs. The HMM has hidden states for deletion and insertion errors to model frameshifts. ESTScan requires species-specific model parameters derived from curated training sequences which could undermine its usefulness for novel transcriptomes. Another tool DECODER (Fukunishi and Hayashizaki, 2001) uses intrinsic features such as the Kozak motif, codon usage, and position of the initiation codon to score coding regions. It also inserts or deletes a nucleotide in all frames to correct for frameshifts. Several SVM-based methods (Kong, et al., 2007; Liu, et al., 2006) were developed to identify transcripts that contain protein-coding genes and discriminate them from non-translatable transcripts. However, those methods do not parse a transcript into coding and non-coding regions.

A recent *ab initio* tool, TransDecoder[3], a companion of the *de novo* transcriptome assembler Trinity (Haas, et al., 2013) identifies putative coding ORFs in reconstructed

[3] https://transdecoder.github.io/

transcripts. It uses intrinsic features such as ORF length and log-likelihood score of a Markov model. It generates the training set for the Markov model by a simple automatic procedure that identifies long open reading frames in the assembled transcripts. Another *ab initio* tool, TransGeneScan (Ismail, et al., 2014), is designed specifically for gene prediction in meta-transcriptomic sequences. It uses the HMM and incorporated hidden states to account for indels that cause frameshifts.

Eukaryotic transcripts in the spliced form share the same characteristics as prokaryotic genomic sequences: both sequences contain intron-less genes. Gene prediction methods of prokaryotic genomic sequences can be applied to eukaryotic transcript. Large volume of transcriptomic data also enables unsupervised training of species-specific parameters for high-order non-homogeneous coding model. The HMM combined with unique Kozak pattern for translation initiation site in eukaryotes can greatly improve the accuracy of identify the precise boundary of coding regions in a transcript. In Chapter 3 we describe the new self-training algorithm for gene prediction in RNA transcripts.

1.3 Prediction of frameshifts in genomic and metagenomic sequences

Frameshifts change the reading frame of protein-coding genes and affect correct gene calling. Similar to gene prediction approaches, methods to predict frameshifts are also classified as two types: similarity-based or *ab initio*.

In similarity-based approaches, the DNA sequence are translated into three frames and searched against protein databases using sequence-alignment methods such as BLASTP. Getting more than one hit mapping to different frames of the sequence

indicates a frameshift or a fusion gene (Claverie, 1993; Posfai and Roberts, 1992). Dynamic alignment algorithms that compare the three translation frames of a DNA against a protein profile accounting for indels can also identify frameshifts (Birney, et al., 1996; Guan and Uberbacher, 1996). Again these methods rely on the existence and quality of the protein databases as well as the quality of alignment.

Ab intio methods leverage intrinsic statistical features of the sequence to identify frameshifts. Again Markov models and hidden Markov models are shown to be very useful. The posterior probabilities of coding potential in all reading frames determined by GeneMark were used to determine the change of frame (Kislyuk, et al., 2009; Medigue, et al., 1999). Several HMM-based approaches explicitly model transitions between coding frames in the hidden states to predict frameshifts (Antonov and Borodovsky, 2010; Schiex, et al., 2003). Among them are tools specifically designed for error-prone sequences such as ESTs (Iseli, et al., 1999) and metagenomic sequences (Rho, et al., 2010). Note that *ab initio* approaches require genome-specific training sequences. Therefore for novel sequences of unknown origin, unsupervised training methods or heuristic models are especially useful. In Chapter 4 we describe a frameshift detection tool that combines HMMs and the heuristic approach to predict genes with or without frameshifts in short metagenomic sequences.

CHAPTER 2
IMPROVED *AB INITIO* GENE PREDICTION THROUGH LOCAL-GC ADAPTATION AND ADAPTIVE TRAINING

Abstract

Although computational prediction of prokaryotic genes is sometimes considered a solved problem, the rate of prediction errors of even the state-of-the-art tools is not negligible. Short genes and gene starts are often cited as difficult to predict; the prediction of genes located in genomic regions with atypical GC composition, e.g. pathogenicity islands, are prone to errors as well. Here we describe a new algorithm and software tool GeneMarkS-2 that improves over previously developed GeneMarkS. At the first step of analysis, the new algorithm employs heuristic models with parameters adjusted to local GC content. In the subsequent iterative self-parameterization GeneMarkS-2 attempts to determine the features of transcription and translation mechanism and makes adaptation of the model structure to the class of genomes defined by these features. The algorithm controls the balance of sensitivity and specificity by selection of the orders of the Markov chain models as well as duration parameters of the generalized hidden Markov model. Genes with 'atypical' codon usage located in compositionally biased regions, such as pathogenicity islands, are particular targets of the new algorithm. The accuracy of GeneMarkS-2 assessed on several test sets was shown to be favorably compared with other state-of-the-art gene prediction tools.

2.1 Introduction

With the exponential growth of the volume of sequence data in genome databases the power of the homology-based methods for gene identification is constantly increasing. Still the whole universe of microorganisms may not be fully described any time soon. New microbial genomes with more than 50% of genes not showing similarities to known protein families continue to appear in sequencing projects. Moreover even in genomes with highest percentage of genes detected by the homology-based methods accurate *ab initio* gene finding is required to complete the annotation process.

In comparison with the task of finding eukaryotic genes with introns and alternative splicing the prokaryotic task looks simpler. The focus in prokaryotes is on prediction of gene overlaps, gene starts, short genes, and genes with atypical composition. Earlier developed tools for gene finding in prokaryotic genomes, GeneMarkS, Glimmer, and Prodigal are sufficiently precise (Besemer, et al., 2001; Delcher, et al., 2007; Hyatt, et al., 2010; Lukashin and Borodovsky, 1998; Salzberg, et al., 1998). The accuracy of predicting the gene location with correct strand, reading frame, and the gene 3' end gets as high as 97-99%, thus with 1-3% of false negative rate; over-prediction, the false positive rate, is harder to assess due to presence of pseudogenes. A more challenging task is to correctly pinpoint a translation start site; the estimated accuracy currently is in the range of 80-90%.

We describe here GeneMarkS-2, a substantially re-designed version of the *ab initio* gene finder GeneMarkS that has been constantly updated since 2001. GeneMarkS-2 has the following new features: (i) The HMM architecture was expanded to account for

possible atypical genes with GC content from 30% to 70%. Emission probabilities for the atypical states were derived by the approach analogous to one used in MetaGeneMark (Zhu, et al., 2010); (ii) We developed and implemented a log-odd score based dynamic programming algorithm that approximates the Viterbi algorithm but is more flexible for adding new features; (iii) We introduced a new adaptive training approach for iterative parameterization of GeneMarkS-2 to improve control of convergence to biologically relevant point in the parameter space; (iv) We developed a modified Gibbs-sampling approach incorporating in the objective function for motif search the distribution of length of the sequence between the conserved motif in the gene upstream region and the translation initiation site; (v) We introduced the classification of genomes into the three types depending on the organization of the gene upstream regulatory regions and developed the three types of models for sequences upstream of gene starts.

The new software tool is favorably compared with the earlier GeneMarkS as well as with other state-of-the-art gene finders in identifying true genes, especially atypical genes, and predicting correct gene starts. GeneMarkS-2 produces the least number of false positive predictions in both real genomic sequences and simulated non-coding sequences.

2.2 Methods and materials

2.2.1 Test sets preparation

2.2.1.1 Test sets of COG genes and non-coding sequences

Genomes of 115 bacteria and 30 archaea were downloaded from the NCBI[4] (the list of species names and RefSeq ID is provided in Supplementary Table 1). This set was spanning 22 bacterial and archaeal phyla, with genomes varied in genome size, type of genetic code, and GC content (Figure 2.1A). To minimize the effects of possible annotation errors, we selected genes whose protein products show evolutionary conservation and as such belong to COGs (clusters of orthologous groups (Galperin, et al., 2015; Tatusov, et al., 2003; Tatusov, et al., 1997), see Figure 2.1B). A 'COG gene' missed in prediction was counted as false negative (FN). The false negative rate was calculated with respect to the number of 'COG genes' in the genome. To assess false positive (FP) rate we counted predictions in artificial random sequences. Construction of species-specific models of non-coding regions was done as follows. We masked the genomic regions *annotated* as protein-coding genes, RNA genes, or pesudogenes. The remaining sequences were used to estimate parameters of the second-order Markov chain model. For each species the model generated ten artifical non-coding sequences with length 1Mb each. Notably, the density of random ORFs depended on the GC content (Figure 2.1C).

---

[4] ftp://ftp.ncbi.nlm.nih.gov/genomes/

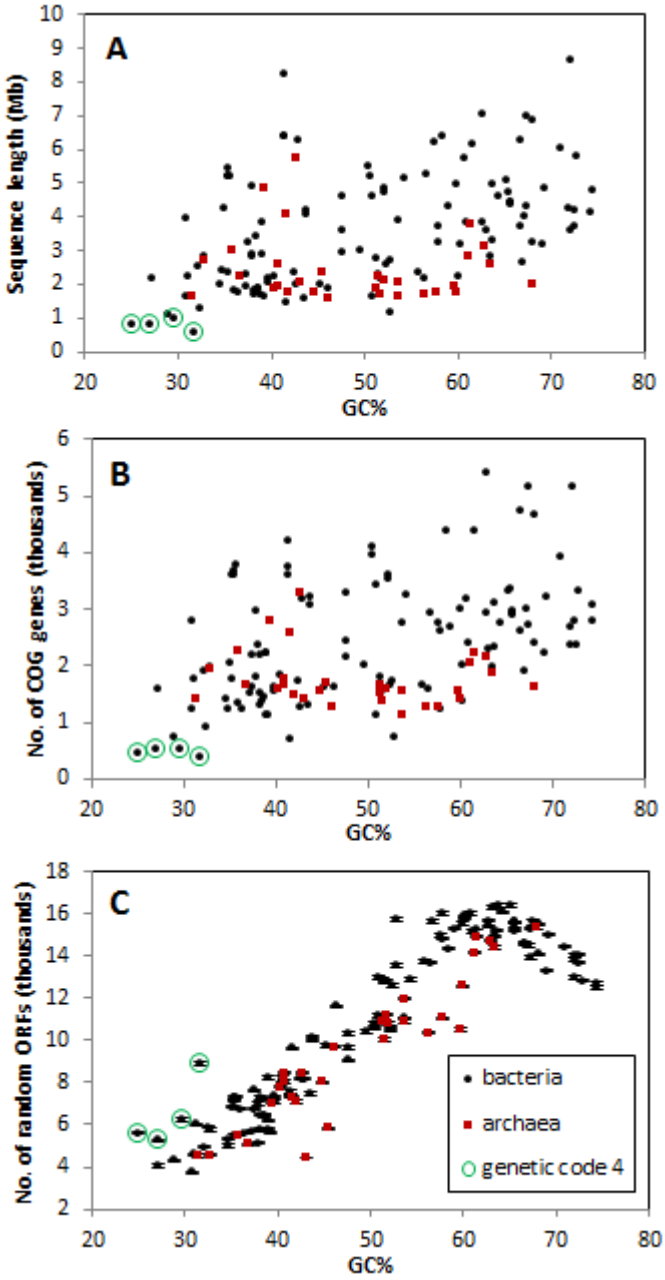

Figure 2.1 Test sets of 115 bacterial and 30 archaeal genomes.
(A) The genome GC content vs. the length of the genomic sequence in each genome. (B) The genome GC content vs. the number of COG-supported genes in each genome. (C) The genome GC content of the original genomic sequence vs. the average number of random ORFs longer than 90nt in non-coding sequences with length 1Mb simulated with genome specific parameters. The standard deviations are shown as upper and lower bars around each dot. Archaeal genomes are depicted in red. Species with genetic code 4 are indicated by green circles. Genetic code of all other species is code 11.

2.2.1.2 Test sets of genes with annotation supported by proteomic data

Data on mass-spectrometry-determined peptides mapped to genomes of 63 prokaryotic species (Venter, et al., 2011) were downloaded from the Pacific Northwest National Laboratory (PNNL). The quality control described in (Venter, et al., 2011) included i/ requirement that peptide/spectrum match to the six-frame translation of the genome with the MS-GF+ software tool[5] would have score with P-value better than 1e-10; ii/ removal of low-complexity peptides; iii/ removal of ORFs lacking a uniquely mapped or fully tryptic peptide, iv/ requirement that a proteomics-confirmed ORF would have at least two matching peptides separated by less than 750nt distance[6]. Peptide data related to five species were not accepted to the tests for the reasons of either too few peptides (<10) or because of the presence of in-frame stop codons in the mapped sequences (Supplementary Table 2). We used the peptide coordinates to find the minimal length ORF that spans the mapped region. There were 1,209,658 peptides mapped to 87,417 ORFs. The selected peptide-supported ORFs (psORFs) made another test set for assessment of accuracy of gene prediction.

2.2.1.3 Test sets of genes annotated in known pathogenicity islands

Sequences of 222 pathogencity islands (PAIs) annotated in 89 genomes were downloaded from the PAIDB database (Yoon, et al., 2015). All 222 islands were previously described in publications. The PAIs were given by coordinates in RefSeq genomic sequences and contained in total 6,748 genes.

---

[5] http://omics.pnl.gov/software/ms-gf
[6] http://omics.pnl.gov/project-data/prokaryotic-proteogenomics

2.2.1.4 Test sets of genes with experimentally verified starts

The N-terminal protein sequencing is a standard but not frequently used technique to validate sites of translation initiation (protein N-terminals and gene starts). Relatively large sets of genes with validated starts are known for the genomes of bacteria *Escherichia coli*[7] (Rudd, 2000; Zhou and Rudd, 2013), *Mycobacterium tuberculosis* (Lew, et al., 2011), *Natronomonas pharaonis*, and *Aeropyrum pernix (Aivaliotis, et al., 2007; Yamazaki, et al., 2006)* (Table 2.1).

Table 2.1 Number of gene starts predicted correctly by the four gene finders in N-terminal verified genes from the six genomes.
The genomes are classified as class one (strong RBS), class two (weak or no RBS), and class three (leaderless transcription with both promoter and RBS signals).

Species	No. of verified genes	Genome class	No. of verified genes predicted correctly			
			GeneMarkS	Glimmer	Prodigal	GeneMarkS-2
*A. pernix**	130	1	126	119	127	127
E. coli	769	1	722	714	751	743
*H. salinarum**	530	3	501	457	514	515
M. tuberculosis	701	1	575	572	620	633
*N. pharaonis**	315	3	310	293	309	307
Synechocystis	96	2	82	79	92	91
Total	2,541		2,316	2,234	2,413	2,416

*Archaeal genomes

[7] http://www.ecogene.org/?q=verified_set

2.2.2 Algorithm design

2.2.2.1 Genome modeling in GeneMarkS-2

The GeneMarkS-2 algorithm uses genome representation as a generalized hidden Markov model (GHMM, also known as HMM with duration or hidden semi-Markov model). The structure of some elements of the GHMM architecture, particularly the order of the Markov chains involved in modeling of protein-coding regions as well as the states for the upstream regulatory regions are selected automatically in the course of adaptive training. The GHMM parameters are derived by iterative unsupervised training.

The GHMM of GeneMarkS-2 (Figure 2.2) expands the GHMM model of GeneMarkS (Besemer, et al., 2001; Lukashin and Borodovsky, 1998). A protein-coding gene is modeled by a group of states including the protein-coding state and the states representing sequences around the gene start (upstream and downstream). We make distinction between the three types of upstream signals: a ribosomal binding site (RBS), a promoter box (in leaderless transcription), or an upstream 'signature' (in case the self-training does not identify strong RBS or promoter signal). The RBS and promoter states emit fixed length 'signal' sequences (e.g. 6nt) described by the positional Markov model followed by a variable length spacer (distance between the gene start and the identified signal); parameters of the signal model and spacer length distribution are determined in self-training. The *upstream signature* state (introduced for the case of weak or no RBS signal) emits fixed length sequence adjacent to the start codon (e.g. 20nt) generated by a positional Markov chain model. We observed the three types of genomes: i/ with all genes preceded by an RBS, ii/ with a subset of genes preceded by an RBS, iii/ with first genes in operons or stand-alone genes preceded by promoter boxes (in genomes with

leaderless transcription) and the other genes with an RBS. In addition to the upstream signals, the gene start model includes the state emitting three nucleotides of the start codon as well as the follow up state emitting the *downstream signature*, a fixed length sequence (e.g. 12nt) generated by a positional Markov chain model.

The protein-coding state has several types, one typical and forty-one atypical. The sequence emitted from the typical or atypical state has variable length described by the gene length distribution fully determined by the gamma function with two parameters. The sequence emitted from a typical state is generated by a three-periodic fourth-order Markov model. The order four does not change during iterations in the main cycle (Figure 2.3), but can be reduced in subsequent adaptive training cycle (see below). Sequences emitted from the atypical state are generated by a heuristic three-periodic fifth-order Markov model (Zhu, et al., 2010). These parameters do not change in iterations.

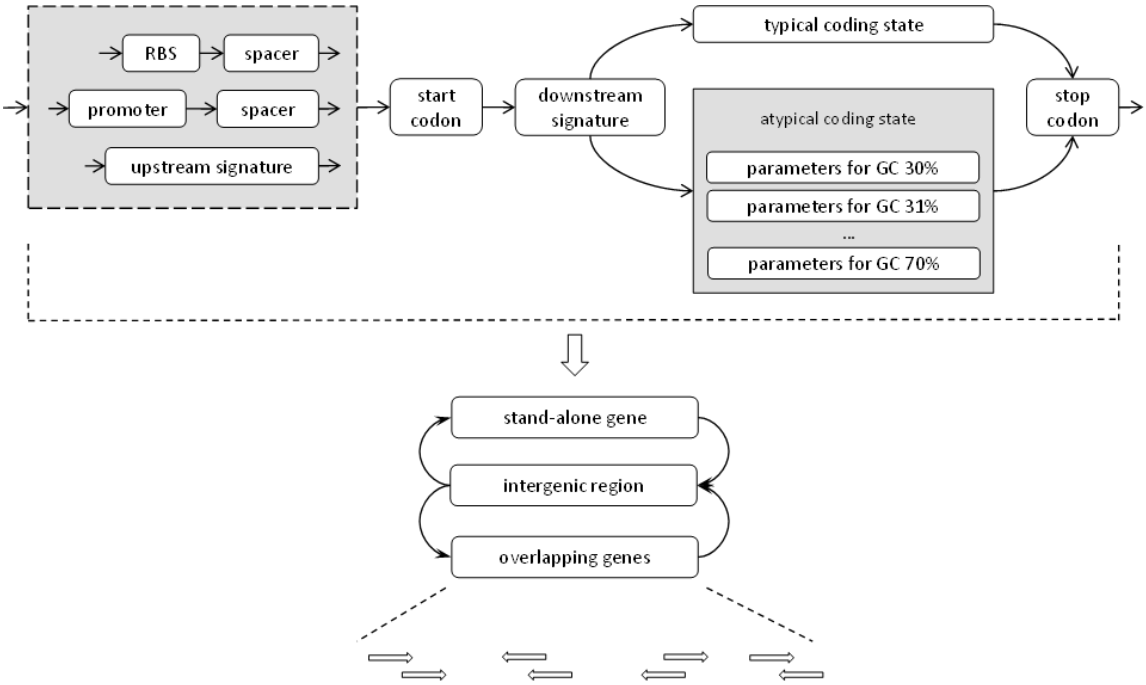

Figure 2.2 State diagram of the GHMM of a prokaryotic genomic sequence as used in GeneMarkS-2.

The arrows designate allowed transitions between the states. Only states for the direct strand are shown. The reverse strand is modeled by an identical set of states with directions of arrows reversed. The reverse strand states are connected to the direct strand states through the intergenic region state and states for opposite strand genes overlaps.

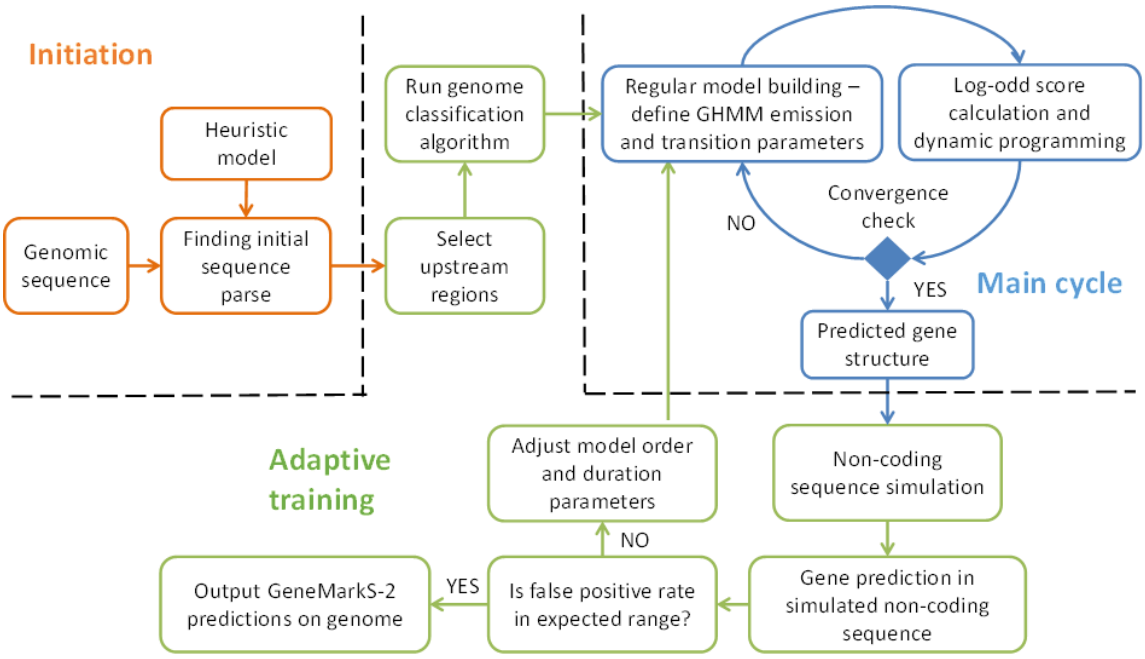

Figure 2.3 Workflow of the GeneMarkS-2 unsupervised training.

2.2.2.2 The self-training algorithm

The GeneMarkS-2 self-training algorithm includes three major steps: initiation, iterative training (the main cycle), and adaptive training (Figure 2.3). In the *initiation* step, the algorithm uses pre-defined heuristic parameters (Zhu, et al., 2010) to create a parse of the input genomic sequence into protein-coding and non-coding (intergenic) regions. The heuristic models of protein-coding regions defined for each '1% GC content bin' from 30% to 70% GC includes the following parameters: 1) transition probabilities of the three-periodic fifth-order Markov model of the protein-coding sequence, 2) transition probabilities of the second-order homogeneous Markov model for the intergenic sequence, 3) the length distributions of protein-coding and non-coding regions, and 4) the frequencies of start and stop codons. Notably, the heuristic parameters are used in MetaGeneMark for gene prediction in anonymous short metagenomic sequences when the genome-specific parameters are not available (Forsberg, et al., 2014; Karlsson, et al., 2013; Nielsen, et al., 2014; Tyakht, et al., 2013). Initial genomic parse done with the heuristic models creates a robust initial training set that demonstrates some advantages over initiations made by Glimmer and Prodigal (Figure 2.4).

After the initial parse of the genomic sequence is determined, the upstream sequences (40nt long fragments adjacent to the predicted gene starts) are selected for the first round of motif search performed by the Gibbs sampling. The results of the motif search are used to classify a given genome into one of the three categories mentioned above.

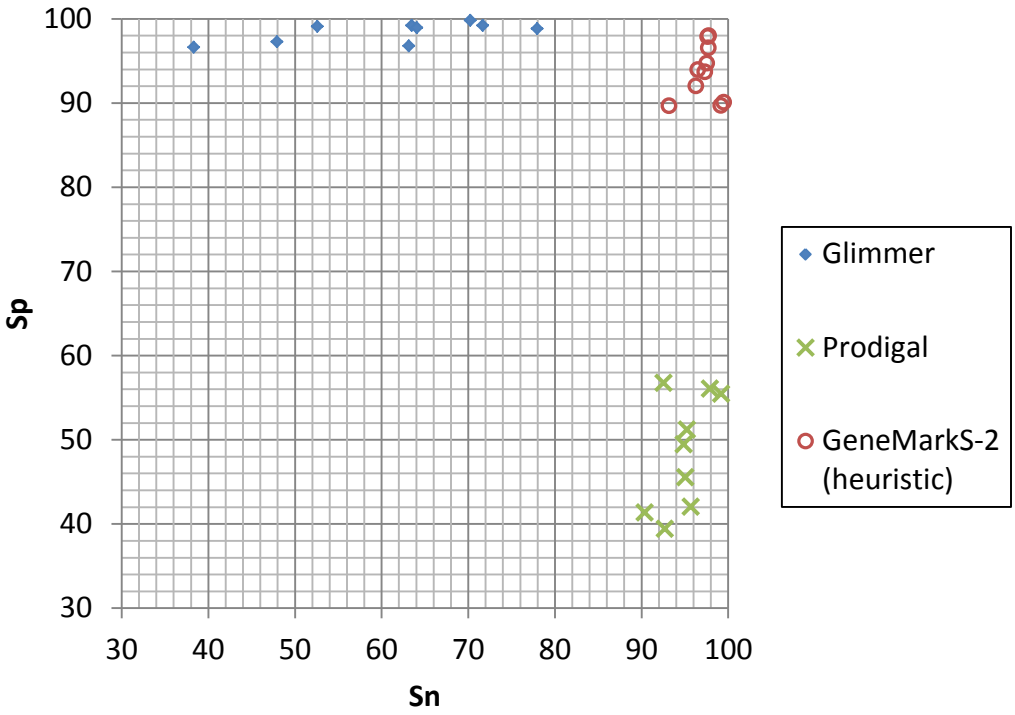

Figure 2.4 Sensitivity (Sn) and specificity (Sp) computed for the initial set of ORFs used in parameter estimation of the three gene finding tools in the ten genomes with whole genome annotation taken as a reference.
The initial sets were obtained as stated above. The ten genomes were from the following species *A. pernix*, *B. subtilis*, *E. coli*, *H. influenzae*, *H. salinarum*, *M. tuberculosis*, *M. genitalium*, *N. meningitidis*, *N. solfataricus* and *Synechocystis*.

The classification criteria are as follows: The discovered motif is considered "strong" if it is localized, i.e., more than 10% of genes have the same most frequent motif-to-start (spacer) length. If the distance from the start to the mode of the spacer length distribution is shorter than 15nt, the motif is classified as a 'strong RBS' (*class one*); otherwise it is classified as a 'promoter box' that appears due to absence of RBS at the first genes of operons and stand-alone genes (*class three*). Genomes without a strong localized motif are classified as *class two*. Representative species of class one, two, or three are *E. coli*, *Synechocystis*, and *H. salinarum*, respectively.

With the classification step finished, the motif finder runs the second time as follows. For a *class one* genome, motif search is limited to the 20nt upstream sequence to build the final RBS model. For a *class two* genome, the motif finder first scans the 20nt upstream sequences of all predicted genes to look for a putative 'RBS word' in the form of hexamer AGGAGG allowing two mismatches. For all genes that contain this 'RBS word' the motif finder builds an RBS model using 20nt upstream sequences. All the other upstream sequences are aligned at the start codon and a positional Markov chain model (*upstream signature*) of length 20nt is built. For a *class three* genome, the motif finder first determines if a gene is i/ the first in operon or a stand-alone gene that is supposed to be preceded by a promoter or ii/ a gene located inside an operon that is supposed to have an RBS. The rational is that in the class three genomes such as the archaeon *H. salinarum*, the first genes in operon and stand-alone genes were observed to have leaderless transcription (Slupska, et al., 2001). Therefore, those genes have a promoter signal close to the gene start (Torarinsson, et al., 2005). Each gene predicted in self-training iterations is classified as first gene in operon or stand-alone gene if the upstream

gene is located in an opposite strand or is located on a distance >22nt; otherwise, it is classified as an internal gene. For the first-in-operon and stand-alone genes the motif finder builds a promoter model using 20nt sequence fragments located between positions -41 to -21 from the gene start. For internal genes, the motif finder builds an RBS model with 20nt long upstream sequences (from -20 to -1 position).

At the gene prediction step instead of the probabilistic Viterbi algorithm to decode the GHMM model, we use a mathematically equivalent dynamic programming approach in the log-odd space, with the log-odd scores computed for each ORF. The score of an ORF is defined as the sum of the start score and the CDS score. For an ORF sequence $x_1x_2 \ldots x_n$ with the start codon $x_1x_2x_3$, stop codon $x_{n-2}x_{n-1}x_n$, GC content ϕ, and length *n*, the start score is defined in equation 6 and the CDS score is defined in Equation 7, in which $x_{-20}x_{-19} \ldots x_{-1}$ denotes the upstream sequence, $y_1y_2 \ldots y_k$ denotes the RBS or promoter motif, *k* denotes the motif length, and M_a denotes the model for state *a*. The last term in equation 7 is the log-odd scores of the durations defined as described in (Lukashin and Borodovsky, 1998); here *C* is a constant depending on parameters *Dc* and *Dn*, the characteristic lengths in the gamma (protein-coding) and exponential (intergenic) length distributions, respectively.

For two overlapping genes *a* and *b* with lengths L_a and L_b, respectively, and the length of overlap *m*, a penalty S_{ovlp} (equation 8) is added to the score. Then the dynamic programming finds the sequence of ORFs and intergenic regions that maximizes the total score in a given iteration.

$$S_{start} = \log \frac{P(\mathrm{x}_1\mathrm{x}_2\mathrm{x}_3 \mid \mathrm{M}_{start\_codon})}{P(\mathrm{x}_1\mathrm{x}_2\mathrm{x}_3 \mid \mathrm{M}_{non})} + \log \frac{P(\mathrm{x}_4 \ldots \mathrm{x}_{15} \mid \mathrm{M}_{down\_signal})}{P(\mathrm{x}_4 \ldots \mathrm{x}_{15} \mid \mathrm{M}_{non})}$$
$$+ \begin{cases} \log \dfrac{P(\mathrm{y}_1 \ldots \mathrm{y}_k \mid \mathrm{M}_{rbs})}{P(\mathrm{y}_1 \ldots \mathrm{y}_k \mid \mathrm{M}_{non})} + \log \dfrac{P(l \mid \mathrm{M}_{rbs\_spacer})}{e^{-\frac{l+k}{D_n}}}, & \textit{if RBS} \\ \log \dfrac{P(\mathrm{y}_1 \ldots \mathrm{y}_k \mid \mathrm{M}_{promoter})}{P(\mathrm{y}_1 \ldots \mathrm{y}_k \mid \mathrm{M}_{non})} + \log \dfrac{P(l \mid \mathrm{M}_{promoter\_spacer})}{e^{-\frac{l+k}{D_n}}}, & \textit{if promoter} \\ \log \dfrac{P(x_{-20} \ldots x_{-1} \mid \mathrm{M}_{up\_signal})}{P(x_{-20} \ldots x_{-1} \mid \mathrm{M}_{non})}, & \textit{if upstream signature} \end{cases} \quad \textbf{(6)}$$

$$S_{CDS} = \log \frac{\max\{P(\mathrm{x}_{16} \ldots \mathrm{x}_{n-3} \mid \mathrm{M}_{typical\_coding}), P(\mathrm{x}_{16} \ldots \mathrm{x}_{n-3} \mid \mathrm{M}_{atypical\_coding,\phi})\}}{P(\mathrm{x}_{16} \ldots \mathrm{x}_{n-3} \mid \mathrm{M}_{non})}$$
$$+ \log \frac{P(\mathrm{x}_{n-2}\mathrm{x}_{n-1}\mathrm{x}_n \mid \mathrm{M}_{stop\_codon})}{P(\mathrm{x}_{n-2}\mathrm{x}_{n-1}\mathrm{x}_n \mid \mathrm{M}_{non})} \quad \textbf{(7)}$$
$$+ \log \frac{Cn^2 e^{-\frac{n}{D_c}}}{e^{-\frac{n-1}{D_n}}}$$

$$S_{ovlp} = -m \log(1 + \frac{m}{2}(\frac{1}{L_a} + \frac{1}{L_b})) \quad \textbf{(8)}$$

The *main cycle* of iterations of gene finding and parameter re-estimation runs until convergence. Given the first parse of the genome sequence made with heuristic models in the initiation step, all the predicted genes (with exception of the ones shorter than 300nt) are used as a training set to derive the parameters of the 'native' model and to make the second sequence parse (Figure 2.3). Next, with the second parse of the sequence defined, for each predicted ORF the score *S* of ORFs computed by the 'native' model is compared with the score *S'* computed for the same ORF in the prediction by the locally adjusted heuristic model (only positive *S'* are considered). If *S* is greater than *S'* the ORF is included into the further training of the 'native' model; if *S* is smaller than *S'*, the ORF is retained as a potential 'atypical' gene and is excluded from training of either

'native' protein-coding model or a model for intergenic region. With all the predicted ORFs thus classified we proceed with a new iteration of estimation of the parameters of the 'native' protein-coding and the non-coding models. The main cycle stops when less than 1% of predicted ORFs change in comparison with the previous iteration.

After reaching the convergence, the algorithm proceeds with the additional 'adaptive training' step in which the model structure is validated. First, the homogeneous second-order intergenic model derived from the main cycle is used to generate a simulated non-coding sequence of 1Mb in length. Then gene prediction with parameters defined in the main cycle is carried out for the artificial sequence. The percentage of ORFs predicted as genes is compared with a threshold empirically chosen to be 1.6%, which means 1.6% of the total number of ORFs (longer than 90nt) is predicted as genes. An error rate higher than the threshold would indicate some issues with the quality of the model training. This issue may result from insufficient sequence volume to estimate parameters of the fourth-order Markov model, weak start signal, or low relative entropy between models of protein-coding and non-coding regions. Thus, the order of the model of the protein-coding region is reduced by one and a new training cycle starts with the reduced order model, etc. If after two adaptive training steps the model order is reduced to two, yet the error rate is still higher than the threshold, the duration parameter of the atypical (heuristic) models is adjusted. Finally, all the ORFs predicted by the 'native' and 'atypical' (heuristic) models are included into the output list of predicted genes.

2.3 Results

2.3.1 Assessment of gene prediction accuracy on the test sets of COG genes and artificial non-coding sequences

To assess gene prediction performance of GeneMarkS, Glimmer3, Prodigal, and GeneMarkS-2 we used 115 bacterial and 30 archaeal genomes (Supplementary Table 1 and Figure 2.1). We run the four gene finders on each genome with default options (except for the minimal gene length which was set to 90bp for all tools) and recorded the number of the unpredicted COG genes. The percent of missed COG genes of each genome was plotted against the genome GC content (Figure 2.5). The overall false negative rate for predicting COG genes is low for all the gene finders tested, with less than 2% missed COG genes for majority of genomes. GeneMarkS-2 displays the lowest average false negative rate and its prediction performance has least dependence on the GC content (Figure 2.5AB). To assess the differential gene finding accuracy for genes with different lengths, we selected groups of the COG genes with lengths between 90-150nt, 150-300nt, 300-600nt, 600-900nt, or longer than 900nt, respectively, and showed number of missed COG genes in each bin (Figure 2.6A). Glimmer has significant lower false negatives for genes in the length range of 90nt-150nt compared to all the other tools. However, this comes at a cost of a significant increase in numbers of false predictions (Figure 2.6B). For the COG genes in all the other bins, GeneMarkS-2 shows better performance than the other three tools. The overall false negative rate of GeneMarkS-2 in prediction of the COG genes is only 0.3%. Note that in Figure 2.5BD we have shown 'zoomed in' graphs of the error rates for only two gene finding tools, GeneMarkS-2 and Prodigal.

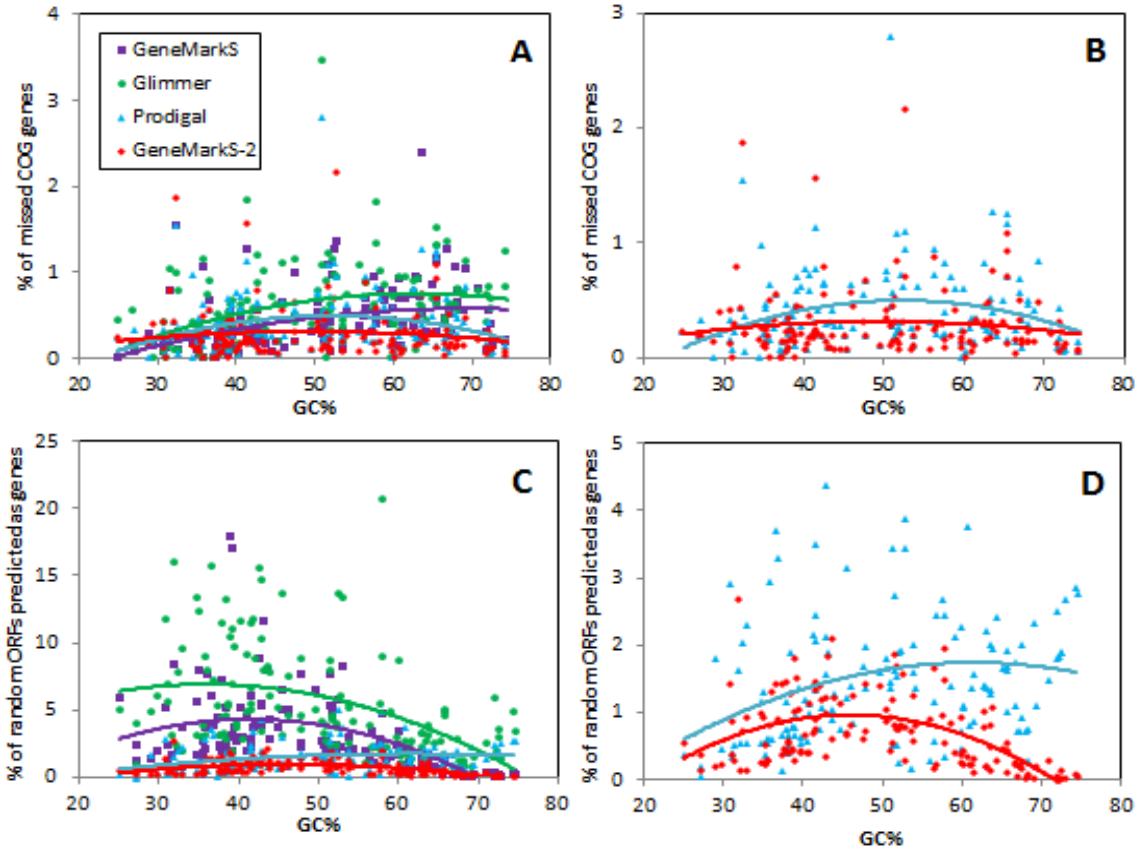

Figure 2.5 Measures of gene prediction accuracy for 115 bacterial and 30 archaeal genomes: % of missed COG genes – panels A and B; and % of random ORFs predicted as genes (panels C and B).
Panels A and C show results for the four gene finders. Panels B and D show results for GeneMarkS-2 and Prodigal in zoomed-in scale in the Y axis.

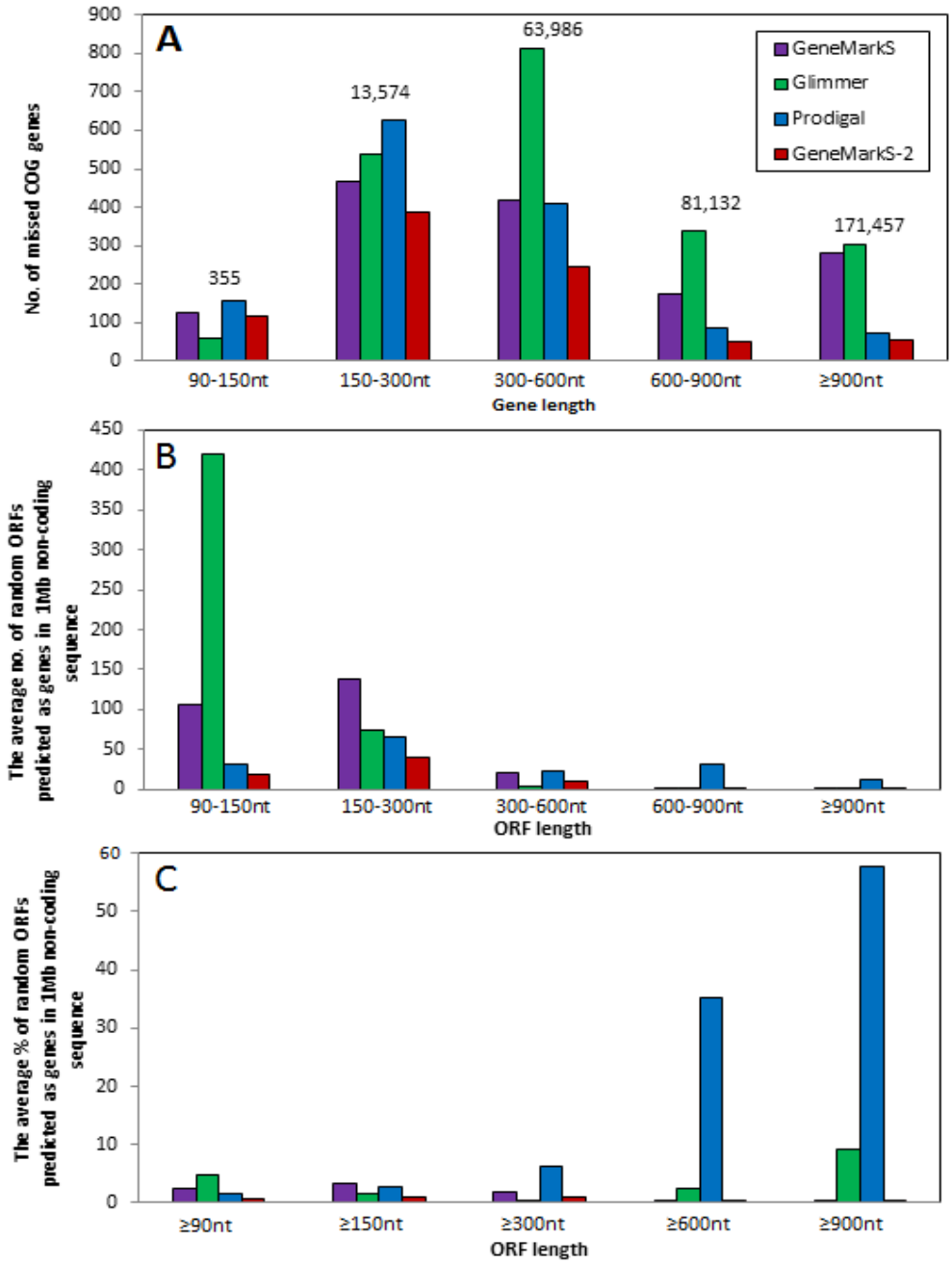

Figure 2.6 Gene prediction accuracy assessed for the four gene finders for 115 bacterial and 30 archaeal genomes.

(A) Number of missed COG genes is shown for five length bins. The total number of COG genes in each bin is indicated above the bars. (B) The average number of random ORFs predicted as genes in 1Mb simulated non-coding sequence in the five length bins. (C) The percentage of random ORFs in simulated non-coding sequence predicted as genes is shown for sets with the minimal length cutoffs.

To assess false positive rates we used sets of species-specific artificial non-coding sequences generated as described in the section 2.2.1.1. Each gene finder was used with parameters estimated for the genome of the corresponding species. The rate of false positive predictions was defined as the ratio of random ORFs predicted as genes to the total number of random ORFs. The numbers of random ORFs longer than 90nt, averaged over 10 simulations of the species-specific non-coding sequence with length 1Mb, are shown in Figure 2.1C. The numbers depend on GC content and are lower for low and high GC genomes while reaching maximum at about 58-65% GC. GeneMarkS-2 was observed to have a significantly lower error rate, e.g. about 50% lower on average than the second best tool, Prodigal (Figure 2.5CD). The increased false positive rate of Prodigal in high GC genomes (Figure 2.5D) may be related to the observed tendency for predicting longer ORFs as genes (see below). A simulated non-coding sequence with high GC (up to 65%) contains more long ORFs than random sequences with lower GC.

To assess the dependence of prediction performance on the ORF length, we grouped ORFs predicted in artificial non-coding sequences into GC bins (Figure 2.6B). Computing the fraction of non-coding ORFs predicted as genes was not quite straightforwad in this case since a predicted ORF might be shorter than the longest ORF in the same location. Therefore, we used a minimal length cutoff and calculated false positive rate as the number of predicted ORFs longer than the cutoff divided by the number of ORFs present in the sequence and longer than the cutoff (Figure 2.6C). GeneMarkS-2 demonstrated consistantly lower error rate than other tools for all the length thresholds. Glimmer perfomed well with the threshold of 300nt, while a large number of shorter ORFs was predicted as genes in the range 90-300nt (Figure 2.6B).

Prodigal, in contrast, has shown an increase in error rate in identification of ORFs longer than 300nt (Figure 2.6C).

2.3.2 Assessment of gene prediction accuracy on the test sets of genes supported by proteomics

Proteomics-supported gene sets (psORFs) were available for 58 species (see 2.2.1.2). We identified the three types of errors: 1) a "missed" or false negative, if the psORF was not predicted; 2) a "wrong" prediction or false positive if the predicted gene overlapped for more than 20nt with a psORF situated in a different strand or frame; 3) a "shorter" gene prediction if the 3' end of a psORF was predicted correctly but the gene start was predicted inside the psORF.

Rather rarely, in 0.41% of cases, we could not locate a start-to-stop psORF with a valid start codon within a stop-to-stop psORF. These observations could occur for the following reasons: i) errors in the mass spectrum generation; ii) errors in the spectrum/peptide mapping inference; iii) mapping of the peptides to pseudogenes; iv) non-canonical features of a gene, e.g. stop-codon read-through or non-canonical start codon. We have estimated the fraction of cases that were due to reasons i) and ii) as 0.16%. Since the single leftmost peptides mapped to an ORF could be erroneous, for the accuracy assessment we also selected sets of psORFs whose start locations were supported by at least 2 or 3 peptides (Figure 2.7).

As the result of the assessment we have observed that GeneMarkS-2 sets of predictions had the least number of "missed" genes and the least number of "wrong" genes (Figure 2.7). In terms of gene start predictions, GeneMarkS-2 and Prodigal made significantly less numbers of "shorter" predictions compared with Glimmer and

GeneMarkS-1. Prodigal produced a slightly lower number of "shorter" predictions; however, the difference in the error rates (0.16%) turned out to be comparable to the estimated error rate of the evaluation method.

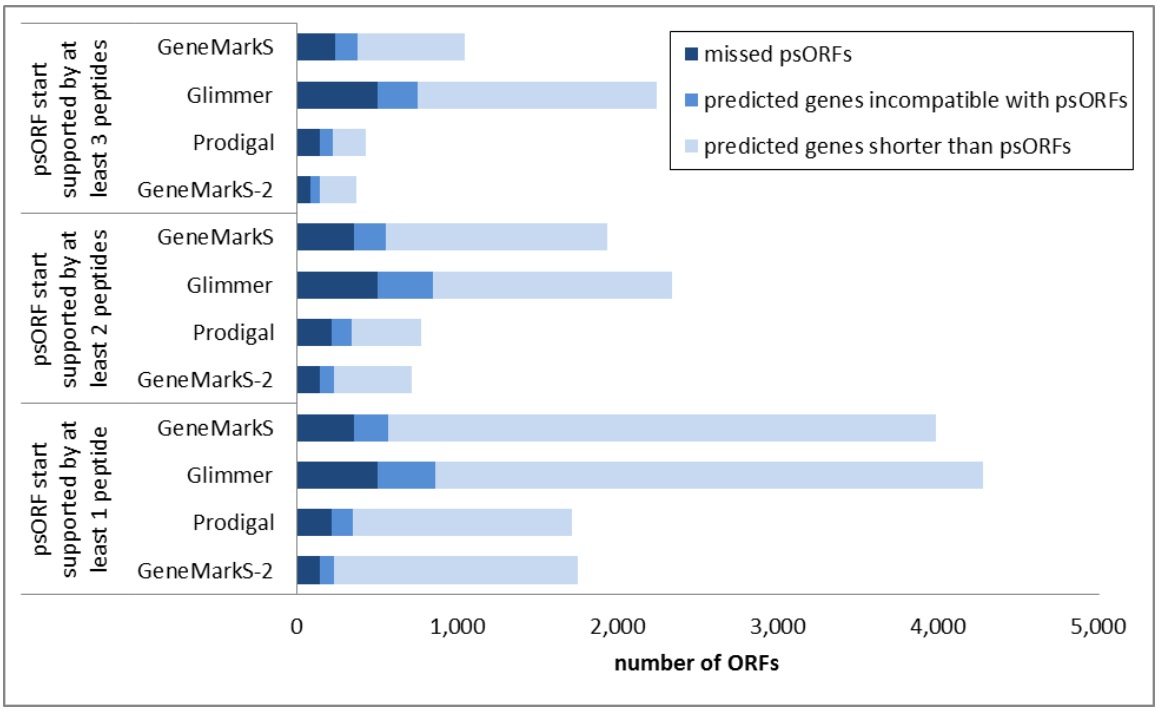

Figure 2.7 Gene prediction accuracy assessed for the four gene finders on peptide-supported ORFs.
Each psORF had to have at least two supporting peptides to be included in the comparison. We recorded 1) psORFs "missed" in predictions; 2) predicted genes incompatible with observed psORFs if a predicted gene overlapped by more than 20nt with a psORF situated in a different strand or frame; 3) genes predicted "shorter" than a psORFs if the 3' end of a psORF was predicted correctly but the gene start was predicted inside the psORF. We show the results for the sets of psORFs whose leftmost starts were supported by at least one, two, or three mapped peptides.

2.3.3 Assessment of accuracy of gene start prediction on the test sets of genes with experimentally verified starts

The gene start prediction accuracy was further evaluated on the sets of genes with verified starts available for the six species (see 2.2.1.4). We observed that the new gene start model of GeneMarkS-2 improved gene start prediction significantly from GeneMarkS. For *class one* genomes such as *E. coli*, the RBS motif became more localized (Figure 2.8AA'). A dramatic change in the outcome of the motif search was observed in the *class one* genome of *M. tuberculosis*; the Gibbs3 method failed to find the RBS motif while MFinder in GeneMarkS-2 detected a very strong RBS (Figure 2.9). For *Synechocystis,* a *class two* genome, GeneMarkS-2 identified 26% of predicted genes as containing RBS. For these genes MFinder detected a strong and well-localized RBS motif (Figure 2.8B'); in contrast, Gibbs3 converged to an apparently random hexamer with a uniform spacer length distribution (Figure 2.8B). In *H. salinarum,* a *class three* genome, GeneMarkS-2 identified 78% of the predicted genes as ones with promoters initiating leaderless transcription. A well-localized model of the promoter motif was built for these genes (Figure 2.8C'). The remaining 22% of predicted genes were identified as having upstream RBS sites (Figure 2.8C''). Overall, GeneMarkS-2 predicted 95% of gene starts correctly, the best performance among the four tools (Table 2.1).

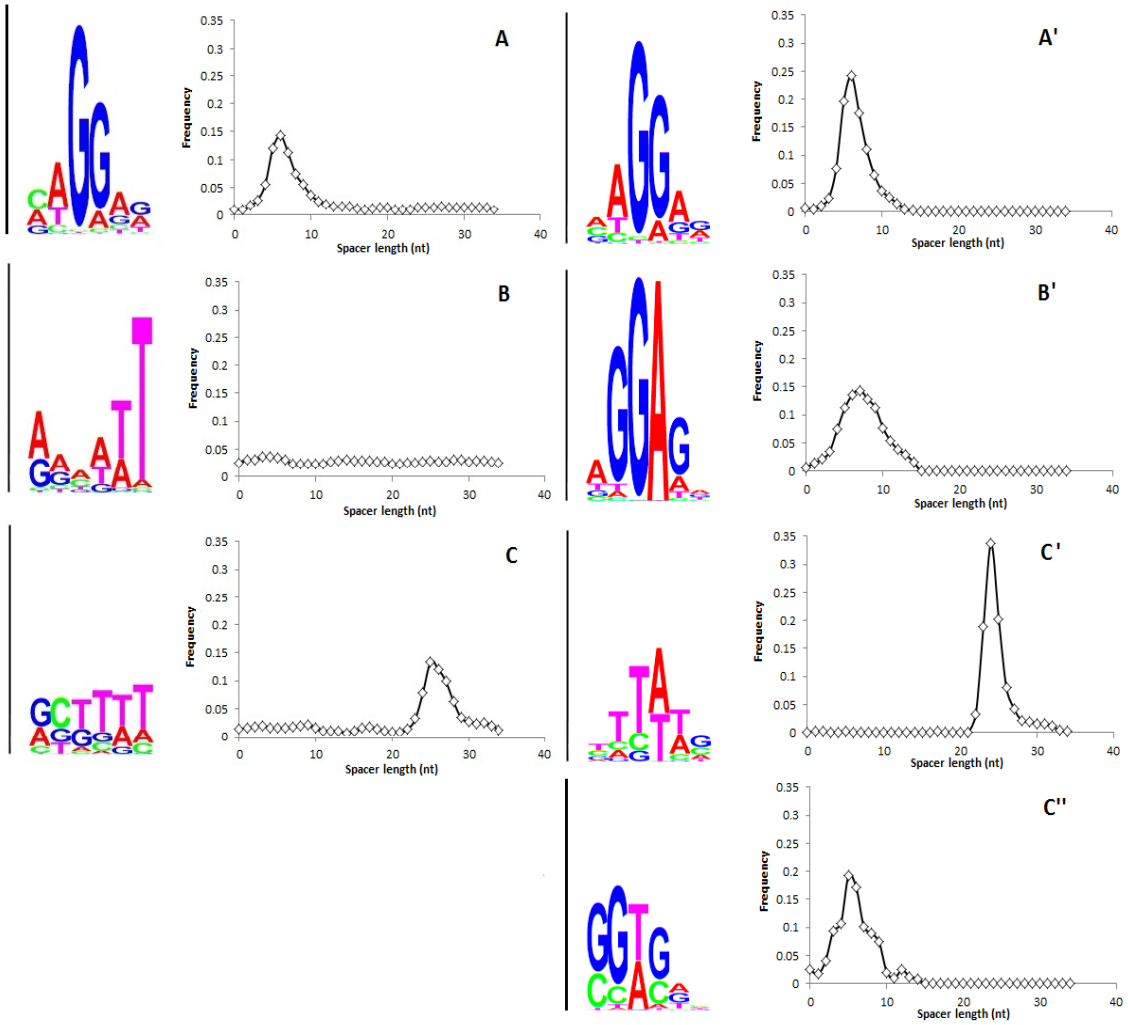

Figure 2.8 Sequence logo and spacer (length between gene start and the identified motif) distribution of motifs detected by GeneMarkS-2 in *E. coli*, a genome of class one (A graphs), *Synechocystis*, a genome of class two (B graphs) and *H. salinarum*, a genome of class three (C graphs).

For the *H. salinarum* genome, the motif finder first divides genes to the first in operons or stand-alone (with a promoter signal) or genes inside operon (with an RBS signal). For first in operons or stand-alone genes the motif finder builds a promoter model (C') using 20nt fragments (located in positions -41 to -21). For all internal genes, the motif finder builds an RBS model (C'') using 20nt sequences located upstream to predicted starts.

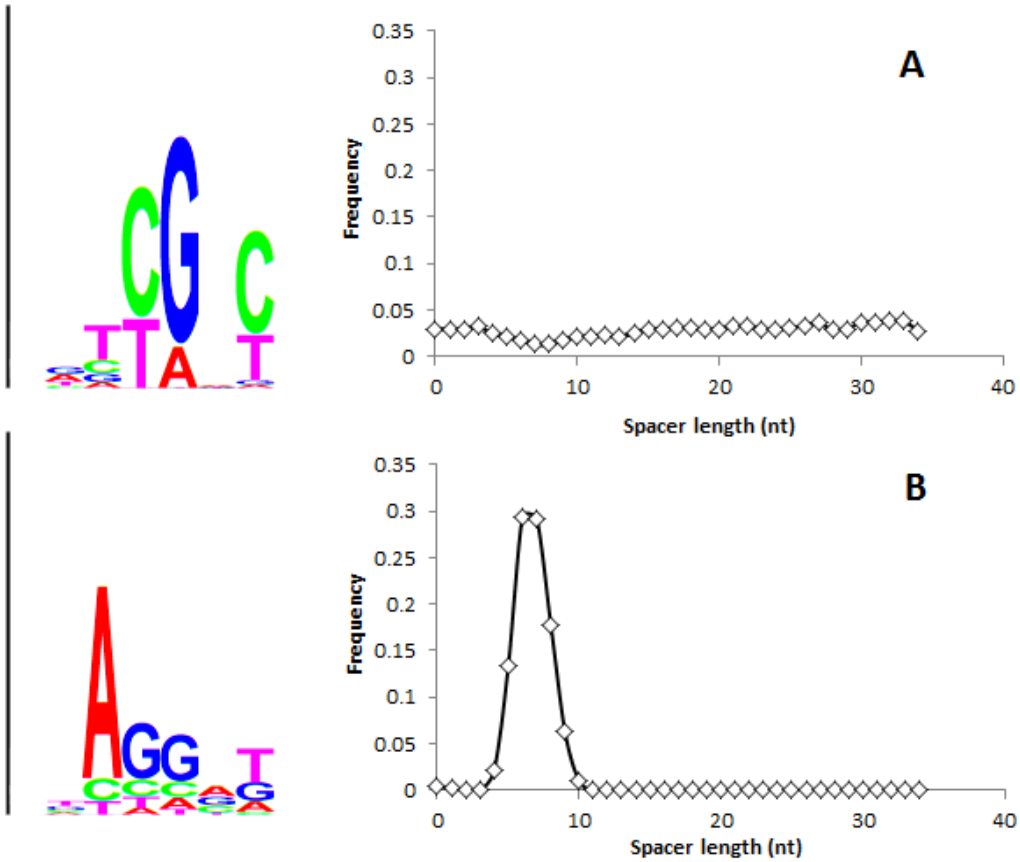

Figure 2.9 Sequence logo of RBS motif model and spacer distribution determined by GeneMarkS (A) and GeneMarkS-2 (B) for the genome of *M. tuberculosis*.

2.3.4 Assessment of accuracy of gene prediction in pathogenic islands

Four gene finders were run on 89 prokaryotic genomes that contain annotated pathogenicity islands (PAIs) with a total of 6,745 genes (see 2.2.1.3). The function was annotated for 3,768 of these genes. These genes are frequently characterized as genes with atypical composition that arguably were horizontally transferred from other species. The number of annotated genes missed in prediction was of interest. GeneMarkS-2 was more sensitive than other tools in predicting genes in the PAIs (Table 2.2).

Table 2.2 Results of the assessment of gene prediction accuracy of the four gene finders in 222 pathogenicity islands (PAIs).
The PAIs contained 6,745 genes with 3,768 functionally annotated.

	No. of missed genes	No. of missed genes with annotated function
GeneMarkS	395	68
Glimmer	424	123
Prodigal	464	72
GeneMarkS-2	399	61

2.4 Discussion

While state-of-the-art *ab initio* gene finders show on average high accuracy in prokaryotic genomes, variability of the prokaryotic genomes still presents a challenge; errors in prediction of atypical genes, short genes, and gene starts are higher than it would be acceptable. GeneMarkS-2 was developed to improve gene prediction in prokaryotes. As shown in the Results section GeneMarkS-2 is the most sensitive among the four tested gene finders i.e. predicts larger numbers of the annotated genes with COG support (Figure 2.5 and Figure 2.6), genes with proteomic support (Figure 2.7), and genes in

pathogenicity islands (Table 2.2). The fraction of missed genes by GeneMarkS-2 is as low as 0.3% for COG genes and 0.2% for genes with proteomic support.

Although GeneMarkS-2 shows high sensitivity in predicting true genes, is does not sacrifice specificity. In fact, GeneMarkS-2 produces lower false positive rates than the other three tested gene finders. In simulated non-coding sequences only 0.6% random ORFs were predicted by GeneMarkS-2 as genes, while the corresponding values for Prodigal and Glimmer were 1.6% and 5%, respectively. Note that the false positive rate of GeneMarkS-2 is uniformly low across the whole range of GC content (Figure 2.5C) and ORFs length (Figure 2.6BC). Correct discrimination of short non-coding ORFs from short genes is a statistically challenging task. However, correct prediction of short non-coding ORFs is important due to their abundance in genomes (Skovgaard, et al., 2001).

An elevated false positive rate in prediction of short ORFs would translate into a large number of erroneous predictions e.g. the case of Glimmer (Figure 2.6B). On the other hand, Prodigal assigns a large weight to the ORFs length; this leads to making 35% and 58% of mis-identification of random ORFs as genes when the ORFs are longer than 600nt or 900nt, respectively (Figure 2.6C). Importantly, the complementary assessment of the gene prediction false positive rate in real genomic sequences in terms of 'numbers of predicted genes incompatible with psORFs' also shows better performance of GeneMarkS-2 in comparison with the other three gene finders (Figure 2.7).

The new locally adjusted heuristic model used in GeneMarkS-2 contributes to the improvement of gene prediction sensitivity. Genes missed by the original GeneMarkS (Figure 2.10A) could be grouped into the three categories: 1) short genes (circled in

blue), 2) long genes with deviated GC content, mostly lower than the genomic one (circled in green) and 3) long genes with the same GC content as the genome (circled in purple). Almost all missed genes in group 2 were recovered by GeneMarkS-2 using the locally adjusted model (Figure 2.10B). These genes indeed are difficult to predict by the native model trained on the genome. Notably, the better prediction of long atypical genes improves overall self-training, as those long genes are removed from the training set of the non-coding model. As a result, prediction of genes in groups 1 and 3 also improved.

GeneMarkS-2 misses only 0.04% of long COG genes (>600nt). Unlike the atypical genes missed by GeneMarkS, most long genes missed by GeneMarkS-2 have GC content similar to the one of the genome (Figure 2.10B); majority of the missed genes are from high GC genomes (Figure 2.10C). A closer examination of those genes revealed that many of them have frameshifts that disrupt the coding frame (Figure 2.10BC). We provide a graph of the protein coding potential for sequence containing for a gene 'GAU_2889' (Figure 2.11) in the *Gemmatimonas aurantiaca* genome (GC% = 64%). In the graph generated by GeneMark (Borodovsky and McIninch, 1993) the high coding potential abruptly moves from frame 1 to frame 3 near sequence position 600, thus exhibiting the pattern typical for the presence of a frameshift. The gene was annotated to have two coding fragments with 3' ends at positions 1000 and 1750 respectively. Annotation of the first fragment includes significant section that would be translated out of frame (between positions 600 and 1000).

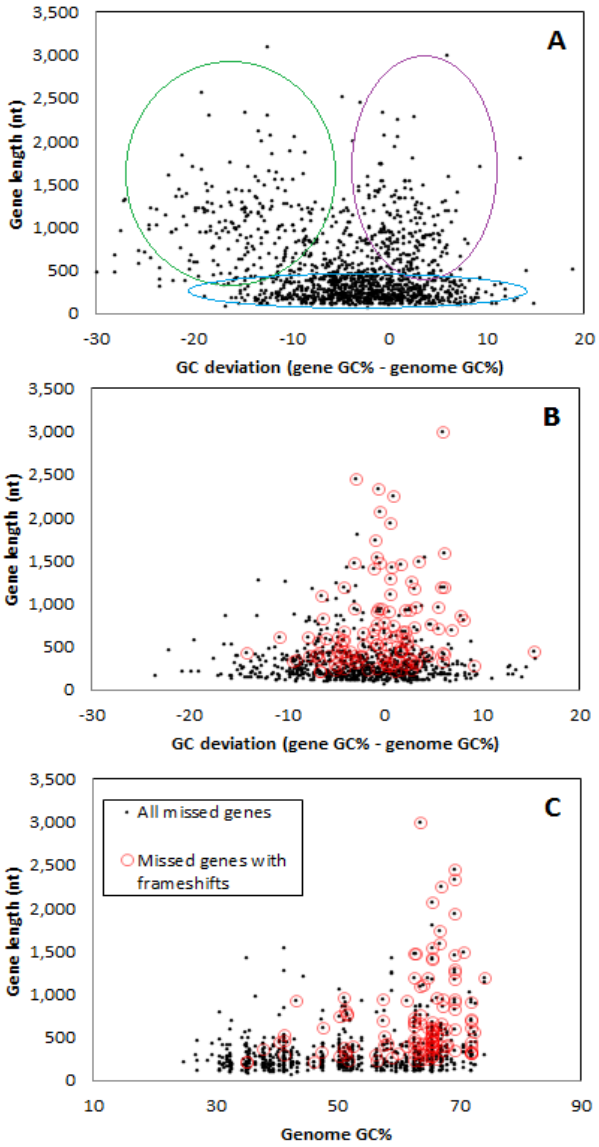

Figure 2.10 Depiction of COG genes (with length >90nt) missed by GeneMarkS (A) and by GeneMarkS-2 (B, C) in 115 bacterial and 30 archaeal genomes.
The X axis in A and B shows the difference in GC content between the genes and the genomic sequence. In A panel genes in the green circle are rather long genes missed by GeneMarkS (with length > 500nt); their composition significantly deviates from the average genome GC content. Genes in the violet circle are rather long missed genes with the GC composition close to average GC content of a corresponding genome. Genes in the blue circle are rather short missed genes. Genes missed by GeneMarkS-2 are shown in panel B in the same X and Y co-ordinates. In panel C the X axis shows the GC content of the corresponding genomes. Dots depicting missed genes where frameshifts were identified are circled in red. Data in panel B shows that GeneMarkS-2 is able to predict long atypical genes missed by GeneMarkS (A). Panels B and C show that missing a long gene is frequently related to frameshifts, which lead to artefacts in prediction (missing gene fragments) with higher frequency in the GC-rich genomes.

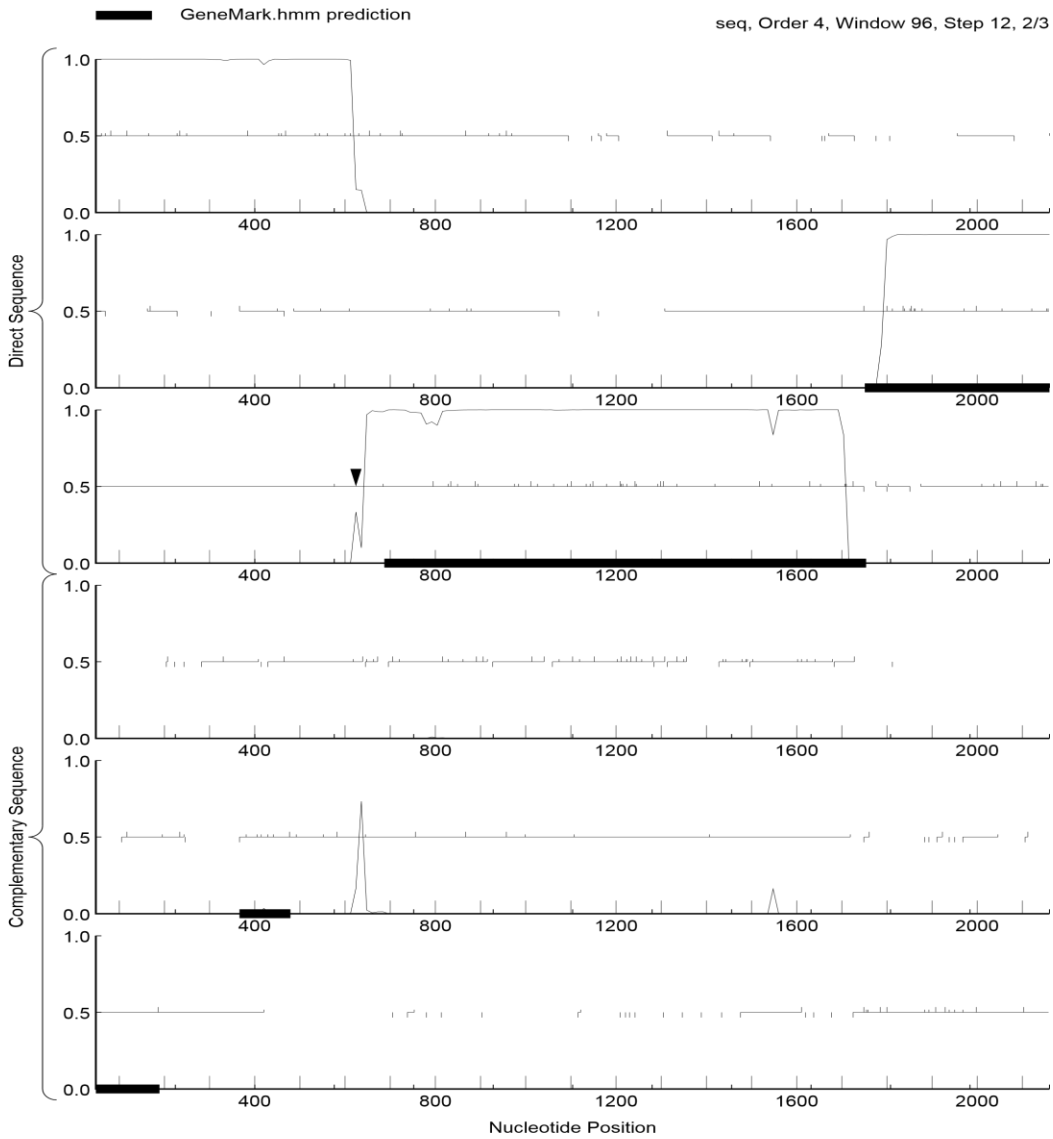

Figure 2.11 The posterior protein-coding probabilities in all six frames are shown along the sequence.

The gene GAU_2889 as annotated has two coding fragments with 3' ends at positions 1000 and 1750 respectively. Annotation of the first fragment includes significant section that will be translated out of frame. The black triangle in frame three indicates a frameshift. Horizontal black bars depict genes predicted by GeneMarkS-2. Genes with frameshifts present a challenge in terms of necessity of annotation of all the fragments with some of them *not ending* at a standard stop codon. The first fragment was not predicted, while the second fragment was. The figure was generated by the GeneMark graphics (Borodovsky and McIninch, 1993)

Genes with frameshifts present a challenge in terms of necessity of annotation of all their fragments with some of them *not ending* at a standard stop codon. While the second fragment was predicted, the first fragment was not and was counted as a "missed" COG gene. To continue this analysis further, we used an *ab initio* frameshift prediction tool MetaGeneTack (Tang, et al., 2013) to analyze all the missed COG genes. We found that 51% of genes longer than 500nt and 62% of genes longer than 1000nt missed by GeneMarkS-2 were likely to contain frameshifts (Figure 2.11BC). Allover, GeneMarkS-2 almost always predicts long genes correctly; a miss or a partial miss of a long gene is likely to indicate a non-canonical feature such as a frameshift.

Out of the 115 bacterial and 30 archaeal genomes in this study, 25 genomes had an estimated error rate higher than the threshold (1.6%) when using the 4^{th} order protein-coding model during the stage of adaptive training. Nine out of the 25 genomes had a non-standard start model: they were either *class two* genomes (with weak RBS motif) or *class three* genomes (with promoter motifs and leaderless transcription). This observation indicated that a better gene start model not only could improve the prediction of gene starts but also could help better predict short genes and eliminate false positives. The RBS scores of the ORFs have significant impact on whether the ORFs would be predicted as genes (Figure 2.12). The correctly defined RBS score moves the total score of a large number of non-coding ORFs into a negative zone, thus eliminating them as gene candidates. At the same time, the RBS score increases the total score of true genes from negative to positive (with only one exception in Figure 2.12).

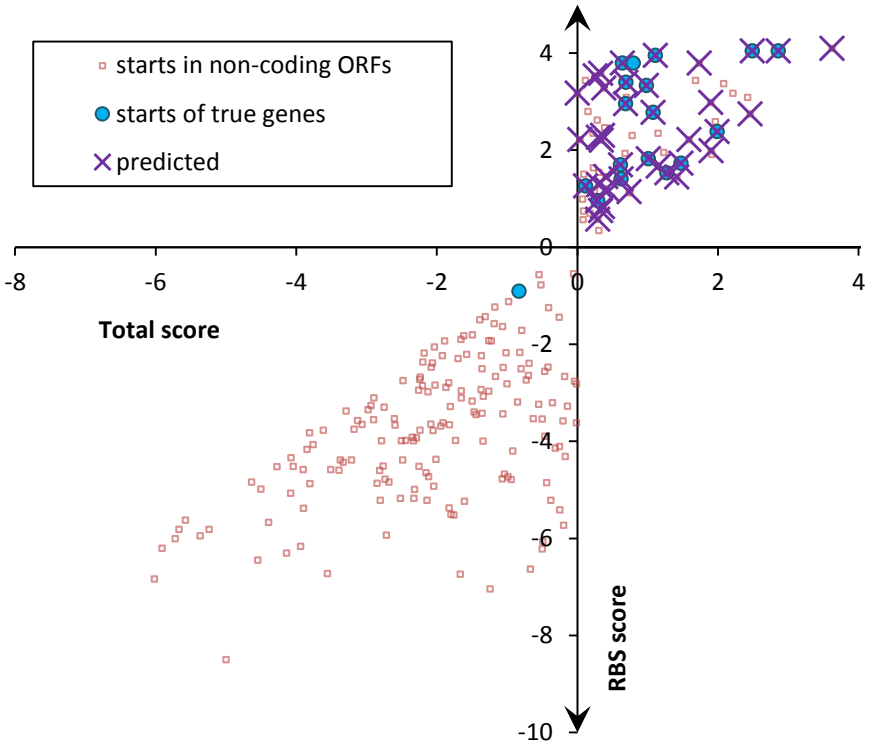

Figure 2.12 ORFs in *E.coli* whose total score has an inversed sign when using RBS. ORFs in the first quadrant have a positive total score with the RBS model but a negative score without it. ORFs in the third quadrant have a negative total score with the RBS but a positive score without it. ORFs with a negative total score will not be predicted while ORFs with a positive score are input to dynamic programming for gene prediction. ORFs that are predicted as genes through dynamic programming are marked in purple.

As demonstrated in Results, GeneMarkS-2 significantly outperformed GeneMarkS in gene start prediction accuracy on the test sets of N-terminal-verified genes from the six genomes (Table 2.1) as well as on the test set of genes confirmed by proteomic data (Figure 2.7). This result is due to better genome-specific characterization of the translation initiation sequence as well as to the improved motif searching algorithm. The new motif finder (MFinder) incorporates length distribution in the object function and thus into the optimization of the motifs alignment.

Classification of the start model into the three categories not only improves gene prediction but also reveals the mechanism of translation initiation at the gene level. For example, our algorithm found 26% of predicted genes in *Synechocystis* to have an RBS and we were able to build a strong and well-localized motif (Figure 2.8B'). Interestingly, the motif consensus sequence is consistent with the previously described 'core' sequence that provided high translation efficiency in *Synechocystis* (Heidorn, et al., 2011). For *H. salinarum* with most genes lacking RBS sites due to leaderless transcription, GeneMarkS-2 was able to detect the promoter box motif (for the first genes in operon) as well as the RBS model (for internal genes) to improve start prediction (Figure 2.8C' and Table 2.1). Thus, our approach is not only able to generate gene predictions, but also able to provide an insight into translation initiation mechanisms of novel genomes.

For initial parameterization of GeneMarkS-2 we used a model with the heuristic parameters (Zhu, et al., 2010). The predicted ORFs served as the "initialization" training set for estimation of the parameters of the "native" model. For the three gene finders, GeneMarkS-2, Glimmer, and Prodigal we compared the initial training sets determined in the ten prokaryotic genomes. For Glimmer's initial training a set of long ORFs is

selected. Although a large percentage of them are true genes, many other true genes are excluded from training which is equivalent to rather low sensitivity in comparison with the whole genome annotation (Figure 2.4). The Prodigal strategy is to use a simple statistics to predict a large set of ORFs as the training set, and then to proceed with the discrimination of some ORFs based on the hexamer statistics and the RBS scores. This strategy of selection of the initial set produces ORFs with high sensitivity but with rather low specificity (Figure 2.4). The ORFs selected by heuristic models (with parameters that do not use any prior knowledge of the given genome other than GC content) immediately provide GeneMarkS-2 with the training set that has high sensitivity and specificity in comparison with the whole genome annotation (Figure 2.4). This analysis shows that the heuristic model is a robust tool for training initialization and for gene prediction per se. Notably, it has been successfully used for gene prediction in short metagenomic sequences.

During adaptive training in GeneMarkS-2 artificial non-coding sequences were used to evaluate the false positive error rate. Note that these sequences are different from those used in the training set: they are simulated from models self-trained by GeneMarkS-2 without information from annotation. We compared the error rate in simulated sequences during training and the actual error rate observed in the test sequences for 115 bacterial and 30 archaeal genomes (Figure 2.13). There is a strong correlation between the estimated error rate and observed error rate ($R^2 = 0.811$). However, the residuals (difference in error rates) get larger as the error rate moves from small to large.

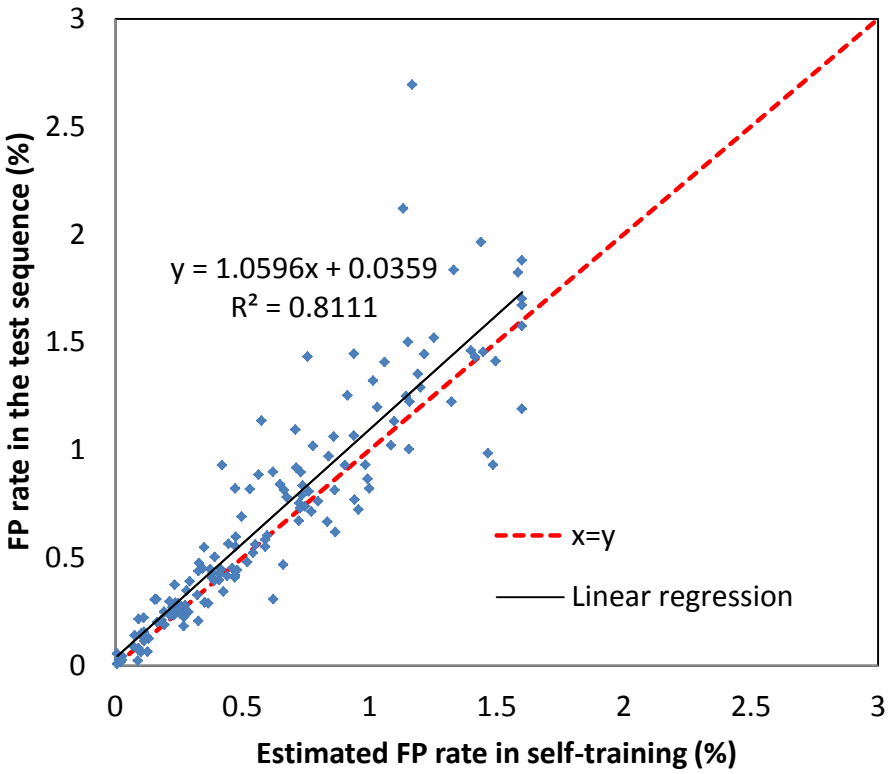

Figure 2.13 Comparison of the false positive (FP) rate during training and testing. During adaptive training, the homogeneous second-order intergenic model derived from the main cycle is used to emit a random non-coding sequence of 1Mb in length. Then gene prediction is carried out on this sequence using dynamic programming. The false positive rate (the percentage of non-coding ORFs in the sequence predicted as genes) is compared with the threshold of 1.6% to determine if reducing the order of the coding state is necessary. To evaluate final gene prediction accuracy, we collected all non-coding sequences from RefSeq annotation to train a second-order Markov model for each genome. This model was used to generate ten artificial random non-coding sequences, each with length 1Mb as a test set. The false positive rate on the training sequence and the average false positive rate in the 10 testing sequences for each genome are shown.

2.5 Software availability

The GeneMarkS-2 software is freely available for academic research and can be downloaded from topaz.gatech.edu/GeneMark/license_download.cgi.

CHAPTER 3
AB INITIO GENE PREDICTION IN RNA TRANSCRIPTS

Abstract

Massive parallel sequencing of RNA transcripts by the next generation technology (RNA-Seq) is a powerful method of generating critically important data for discovery of structure and function of eukaryotic genes. The transcripts may or may not carry protein-coding regions. If protein coding region is present, it should be a continuous (spliced) open reading frame. Gene finding in transcripts can be done by statistical as well as by alignment-based methods. We describe a new tool, GeneMarkS-T, for *ab initio* identification of protein-coding regions in RNA transcripts assembled from RNA-Seq reads. Unsupervised estimation of parameters of the algorithm is an important feature of GeneMarkS-T. It makes unnecessary several steps in the conventional gene prediction protocols, most importantly the manually curated preparation of training sets. We demonstrate that i/ the GeneMarkS-T self-training is robust with respect to the presence of errors in assembled transcripts and ii/ accuracy of GeneMarkS-T in identifying protein-coding regions and, particularly, in predicting gene starts compares favorably to other existing methods.

3.1 Introduction

Prior to the advent of next generation sequencing (NGS), transcriptome data were scarce and limited to full mRNA and EST libraries covering at best a few hundred genes of a given species (Garber, et al., 2011). The RNA-Seq technology (Wang, et al., 2009) generates a vast number of short reads that demanded procedures for assembly of single reads into complete transcripts. Many methods were developed to reconstruct full length transcripts. The reconstruction quality by a number of assembly tools was evaluation by the international RNA-seq Genome Annotation Assessment Project (RGASP) consortium (Steijger, et al., 2013). The important next step in transcript downstream analysis is the transcript annotation, particularly identification of protein-coding regions.

Finding genes in transcripts by mapping known proteins can be successfully implemented only if the protein products of encoded genes have homologs in existing protein databases. Discovery of novel genes requires methods that are alignment-free. Earlier developed *ab initio* gene prediction methods for EST and cDNA sequences, such as ESTscan (Iseli, et al., 1999), used HMMs and required curated training sequences for estimation of model parameters. The supervised training protocol adds downtime that makes application of such tools less practical. The SVM-based method CONC (Liu, et al., 2006) was developed to identify transcripts that contain protein-coding genes and discriminate them from non-translatable transcripts. Since CONC does not parse transcripts into coding and non-coding regions we were not able to use this method in comparisons of gene prediction tools where we have to compare predicted gene borders. A recent *ab initio* tool, TransDecoder, a companion of the *de novo* transcriptome

assembler Trinity (Haas, et al., 2013), generates the training set by a simple automatic procedure that identifies long open reading frames (ORFs) in the assembled transcripts.

Self-training has already been used in algorithms for *ab initio* gene finding in prokaryotic genomes, particularly in the frequently used GeneMarkS (Besemer, et al., 2001), Prodigal (Delcher, et al., 2007; Hyatt, et al., 2010), and Glimmer3 (Delcher, et al., 2007; Hyatt, et al., 2010). Those tools were developed for prokaryotic genomes, but they can be used to predict intronless genes in eukaryotic transcripts.

Here we present a new algorithm, GeneMarkS-T that extends the ability of GeneMarkS to predict prokaryotic genes to identification of continuous (intronless) protein-coding regions in eukaryotic transcripts assembled from RNA-Seq reads or generated by Sanger technology (EST or cDNA sequences). For both biological (e.g. presence of alternative isoforms) as well as technological reasons (e.g. erroneous assembly) several protein-coding genes could be predicted in a single transcript. However, we assume that a correctly spliced and reconstructed eukaryotic transcript should carry a single functional protein-coding gene. Two or more genes in a single transcript would make an operon structure typical for bacteria. With few exceptions eukaryotes possess no operon organization. When two or more protein-coding regions are predicted, GeneMarkS-T assigns a log-odds score to each prediction. We show that the gene with the max log-odds score in a given correctly assembled transcript has a high likelihood to be the true gene.

Transcriptomes of large eukaryotic genomes may exhibit significant variation in nucleotide composition. This inhomogeneity complicates algorithm training and affects

the accuracy of gene prediction. This difficulty can be addressed by clustering the whole set of transcripts based on GC content and deriving several cluster-specific models of the protein-coding regions.

Accurate identification of the translation initiation site (TIS) is not a simple task. Although it is often assumed that the 5'-most AUG codon in a protein coding ORF serves as the true TIS, this is not always the case. True TIS sites were shown to appear in the sequence context known as the Kozak pattern (Kozak, 1987) with relatively weak positional preference for certain nucleotides around the AUG codon. Assessment of accuracy of TIS predictions requires a sufficient number of genes with experimentally verified TIS positions. The recently introduced ribosome profiling, the Ribo-seq technique (Ingolia, et al., 2009) makes it possible to generate large sets of genes with verified TIS positions. This technique uses deep sequencing of mRNA fragments protected by initiating ribosomes (Lee, et al., 2012) to generate a profile of TIS positions. Such dataset can be used as test sets to determine accuracy of TIS predictions.

3.2 Methods

3.2.1 The GeneMarkS-T algorithm design

The GeneMarkS-T and GeneMarkS (Besemer, et al., 2001) algorithms share several parts: i/ the heuristic method of initialization of the hidden semi-Markov model (HSMM) parameters (Besemer and Borodovsky, 1999), ii/ the Viterbi algorithm that finds maximum likelihood parse of transcript sequence into coding and non-coding regions, and iii/ the concept and the method of iterative self-training (Besemer, et al., 2001).

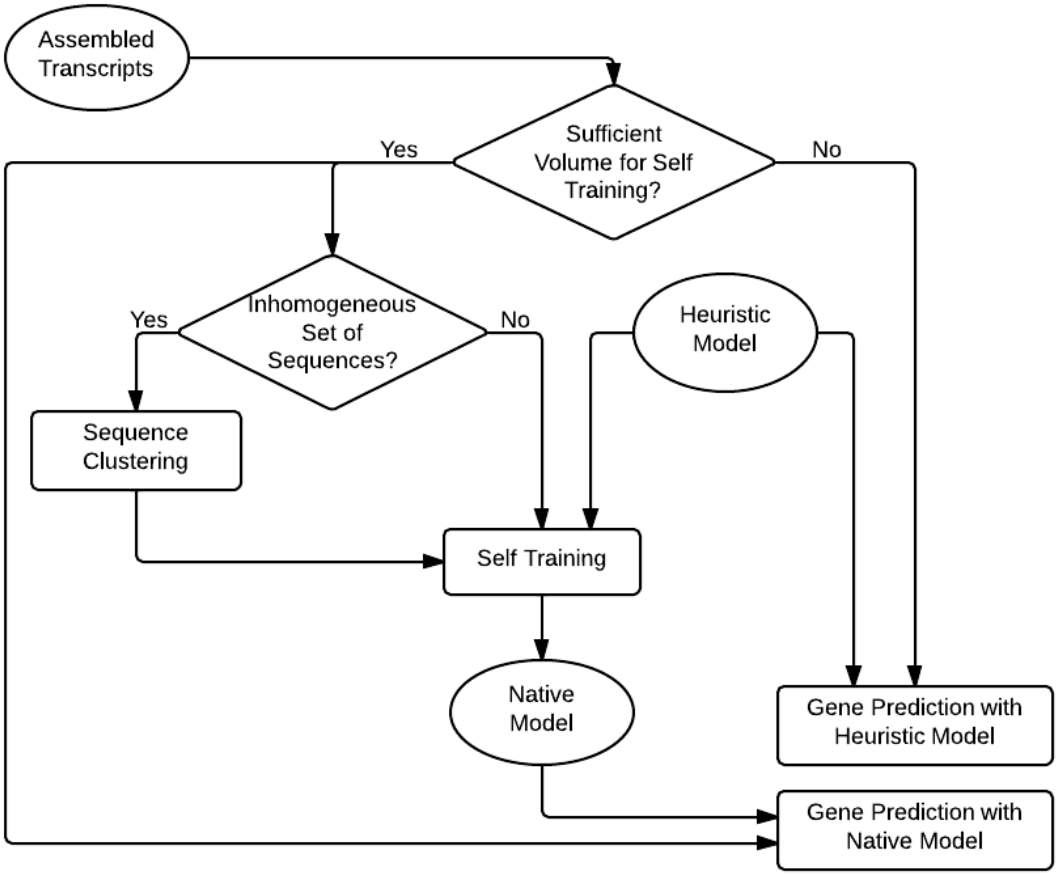

Figure 3.1 Flowchart diagram of the training and prediction steps in GeneMarkS-T.

Important differences between GeneMarkS-T and GeneMarkS are as follows. Contrary to rather short prokaryotic genomes with rather homogeneous GC content across genome, variation in local GC content across much longer eukaryotic genomes may reach 30-40%. Several groups have shown that genomic sequence GC content is one of the major factors driving the genome-wide pattern of codon usage (Besemer and Borodovsky, 1999; Chen, et al., 2004). Therefore, the first step of GeneMarkS-T is clustering transcripts by GC content (Figure 3.1). The number of clusters is determined with respect to the width of the GC composition distribution in the whole set of transcripts. The precise GC borders of clusters are adjusted automatically to place the same volume of sequence in each cluster. The iterative self-training in each cluster runs similarly to that described for GeneMarkS (Besemer, et al., 2001). The procedure starts with initialization of the cluster-specific heuristic model (Besemer and Borodovsky, 1999). Then rounds of predictions of protein-coding regions, selecting new set of sequences of predicted genes for training and re-estimation of parameters follow until convergence, i.e. the set of predicted genes in the current (final) iteration is the same as in the previous iteration. If the total length of transcript data is not large enough for self-training, the heuristic parameters serve as the final parameters and predictions made with heuristic parameters are accepted as final (Figure 3.1). The rationale is the earlier demonstration that heuristic parameters give sufficiently accurate predictions of continuous protein-coding regions in short prokaryotic sequences, e.g. in metagenomic sequences (Besemer and Borodovsky, 1999; Zhu, et al., 2010).

GeneMarkS-T uses an iteratively derived positional frequency model of the sequence around eukaryotic TIS known as the Kozak pattern (Kozak, 1987). The matrix of frequencies is determined from the multiple sequence alignment of 12nt-long fragments surrounding predicted gene starts with nucleotide A situated in position seven.

Recently introduced strand-specific RNA-Seq technology (Vivancos, et al., 2010) carries information on the DNA strand that served as a template for transcription. If this information is available GeneMarkS-T changes the hidden semi-Markov model architecture to reduce the rate of false positive predictions. The hidden states of HSMM standing for the protein coding regions situated in the non-transcribed DNA strand are effectively eliminated. In what follows GeneMarkS-T with the strand-specific HSMM modification is designated as GeneMarkS-T(S).

In each analyzed transcript GeneMarkS-T scores all predicted, complete or incomplete, continuous protein-coding regions and selects the one with the highest score. This score is calculated as the log-odd ratio of the probability of the sequence goes through the coding or non-coding hidden state. For computing the log-odd scores we used the trained models of protein-coding and non-coding sequences as well as their length distributions. The distribution of lengths of protein coding region is modeled as the gamma distribution while the distribution of length of non-coding sequences is modeled by the exponential distribution (Lukashin and Borodovsky, 1998).

3.2.2 Test set preparation

We have prepared the set of *'complete' reference transcripts* from protein-coding mRNA sequences of *A. thaliana*. *D. melanogaster*, *M. musculus*, and *S. pombe*. We downloaded from the RefSeq database mRNA sequences with accession numbers that begin with the prefix "NM_" indicating curated RefSeq records. We removed records with no annotation for start or stop codon, with frameshifts in annotated protein coding regions, or with stop codon read-through. We also removed records with no annotated untranslated regions (UTRs), which is a strong sign that the record was generated from computational gene prediction rather than from experimentally observed RNA transcript. We removed mouse and fly transcripts representing alternative isoforms with the same annotated function; only one isoform, selected at random, was kept per gene. The numbers of downloaded RefSeq sequences and the numbers of sequences in the final set of 'complete' reference transcripts are shown in Table 3.1.

Table 3.1 Composition of the test sets of 'complete' reference transcripts.

Species	No. of mRNAs in RefSeq database*	No. of records with "NM_" prefix	No. of transcripts after filtering (see methods)
S. pombe	5,123	4,841	4,655
M. musculus	77,925	28,887	18,937
D. melanogaster	30,264	30,264	13,241
A. thaliana	35,173	35,173	28,805

* Data were downloaded in October 2014.

For computational experiments with assembled transcripts we generated five sets of *assembled transcripts* of *D. melanogaster* using the following approach. The authors of the latest comprehensive study of the accuracy of transcript reconstruction from RNA-Seq reads (Steijger, et al., 2013) used several tools including Cufflinks (Roberts, et al., 2011), Augustus (Stanke, et al., 2006), Velvet (Zerbino and Birney, 2008), Oases (Schulz, et al., 2012), and Exonerate (Slater and Birney, 2005). The authors of (Steijger, et al., 2013) made available to us the genomic co-ordinates of the exons of assembled transcripts which we used to "splice" together the sequences of transcripts assembled by the five tools mentioned above and analyzed in the course of previous research work (Steijger, et al., 2013). Additionally we constructed a set of 24,804 reference transcripts of *D. melanogaster* from co-ordinates provided by the authors of (Steijger, et al., 2013) who used the FlyBase genome annotation (FB2013_01). We removed from this set 350 transcripts with annotation indicating incomplete genes or some non-canonical features (frameshifts or stop codon read-through), 70 pseudo genes, and 786 non-protein-coding RNAs (ncRNA, tRNA, snoRNA, *etc*.). The final set contained 23,598 reference transcripts that were used for comparison with assembled transcripts to assess the accuracy of both sequence assembly and gene prediction.

We also prepared a set of 'partial' reference transcripts to simulate incomplete transcripts reconstructed from RNA-Seq reads. To come up with a realistic dataset we have analyzed the structure of transcripts observed in sequencing experiments. First, we aligned the transcripts of *D. melanogaster* assembled by the five assembly programs mentioned above to the *D. melanogaster* reference transcripts used in (Steijger, et al., 2013). Next, we determined the relative frequency of presence of reference fragment

parts in the assembled transcripts (Figure 3.2). This analysis indicated that it is common to observe partial transcripts depleted on both ends. Therefore, we simulated partial transcripts by taking complete transcripts with 10% of sequence trimmed in each end.

For accuracy assessment of translation initiation site prediction, we used information of the TISs of protein coding regions in the mouse transcripts verified by the Ribo-seq experiments (Lee, et al., 2012). We used a conservative approach and selected 1,455 transcripts that had only one Ribo-seq identified TIS which matched the annotated TIS; in this set genes longer than 300bp were observed in 1,392 transcripts.

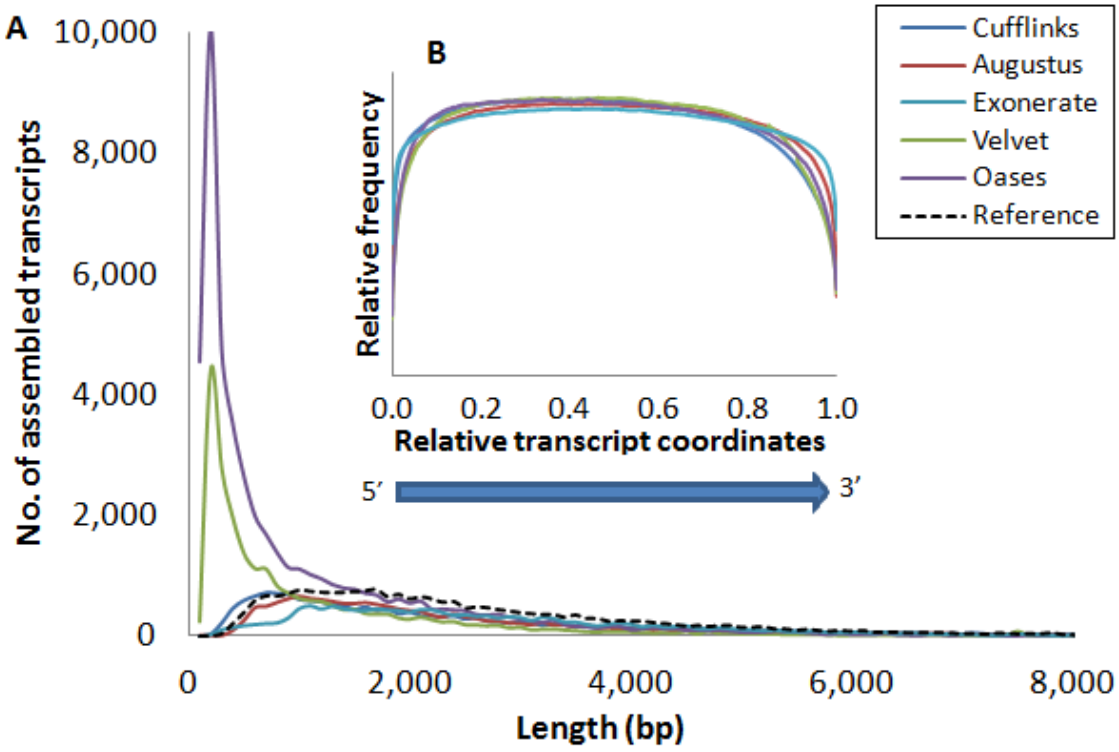

Figure 3.2 Length distribution of reference and reconstructed transcripts.
(A) Length distributions of reference *D. melanogaster* transcripts (RefSeq transcripts) as well as transcripts reconstructed from RNA-Seq reads by Cufflinks, Augustus, Exonerate, Velvet and Oases; (B) Frequency of observing particular transcript section being present in assembled transcripts (shown in relative transcript coordinates). Here top values of the relative frequency are close to 1.0. Note that this graph should not have an integral under the curve equal to one as the transcript sections are arguments for separate random variables (present or not-present).

3.2.3 Aligning Assembled and Reference Transcripts

We used BLASTn to align the *D. melanogaster* transcripts reconstructed by the five methods (Cufflinks, Augustus, Velvet, Oases, and Exonerate) to the reference transcripts. The alignment threshold E-value was set to 0.001. Note that both the assembled transcripts and the reference transcripts were given in the form of exon co-ordinates (annotation) on the genomic sequences. To get reference and assembled transcripts the exons were spliced from genomic sequences in Flybase (FB2013_01). Therefore, in the alignments we required 100% nucleotide identity.

An assembled transcript was classified as 'concordant' if it had a section that could be aligned without gaps to the whole coding region (or to its continuous part) in the reference transcript (Figure 3.3, a-c). The alignment was not attempted to be made in the UTR sections of reference transcripts. Still the requirement was that the 'UTR section' of assembled transcript (situated upstream or downstream of the 'coding' section aligned to the reference transcript coding region) would not be longer that the reference UTR by 300bp (Figure 3.3c). If an assembled transcript did not have a section that could be aligned without gaps to annotated coding regions of reference transcripts (Figure 3.3, d-f), or the 'UTR section(s)' of assembled transcripts was longer than the reference UTR(s) by 300bp (Figure 3.3g), the assembled transcript was classified as 'conflicting'. Assembled transcripts that could not be aligned to references with E-values better than 0.001 were classified as 'not-aligned'.

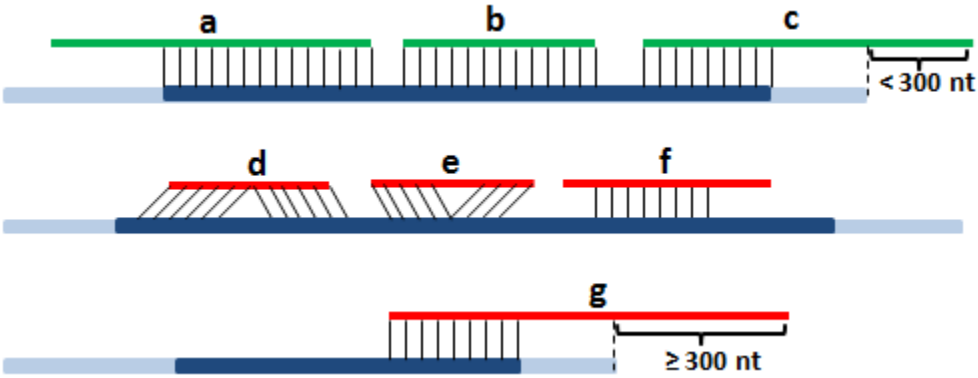

Figure 3.3 Examples of concordant (green) and conflicting (red) transcript assemblies.
'Concordant' transcripts have ungapped BLASTn alignments to reference CDS (dark blue) (a-c); extension beyond the limit of reference UTR is not longer than 300bp (c). 'Conflicting' transcripts are those that contain gaps in alignment to reference CDS (d-f) or/and have a long extension (> 300bp) beyond reference UTR (g).

3.2.4 Assessment of Gene Prediction Accuracy

Along with GeneMarkS-T we assessed performance of ESTscan 2.1 (Iseli, et al., 1999) and TransDecoder (http://transdecoder.sourceforge.net), specialized tools for gene prediction in transcripts, as well as Prodigal (Hyatt, et al., 2010) version 2.60 used in the model of prediction "intronless genes".

The accuracy of gene prediction in the test sets was determined by comparison with annotation. A prediction that correctly identified the reading frame was treated as a true positive prediction (TP); a correctly predicted reading frame would entail an exact match between predicted and annotated stop codons (for genes complete on the 3' end). Sensitivity (*Sn*) and specificity (*Sp*) of a set of predictions was computed as *Sn=#TP/(#TP+#FN)* and *Sp=#TP/(#TP+#FP)*, respectively, where *#FN* stands for the number of false negative and *#FP* stands for the number of false positive predictions.

We classify a prediction as "false positive" if it does not match the annotation (in terms of match between the predicted and annotated stop codons). Notably, computational science operates with true sets and false sets to evaluate classification algorithms. This approach is difficult to implement in full in genome analysis and, particularly, in gene prediction. We do use the true set, the set of annotated genes. However, we do not have a verified set of "non-genes". It is difficult to prove experimentally that a particular segment of a nucleotide sequence is not expressed as a part of a protein coding gene. Therefore, what we use essentially as a surrogate "non-genes" are the sequences of open reading frames that are not annotated as genes.

In the test runs, all the parameters of each program were set to default values except for the threshold defining the shortest length of predicted gene. The threshold influences the balance between *Sn* and *Sp*; the shortest length of predicted gene was varied to generate ROC-like dependencies. If the minimum gene length was not among adjustable program settings, as in Prodigal, predicted genes shorter than the selected threshold were filtered out in post-processing. GeneMarkS-T and TransDecoder have standard "strand specific" options for analyzing transcripts generated by assembly of stranded RNA-Seq reads. To emulate such an option for Prodigal we filtered out protein-coding regions predicted in the designated complementary strand.

3.3 Results

3.3.1 Accuracy of Gene Prediction in RNA Transcripts

GeneMarkS-T, Prodigal, TransDecoder, and ESTscan were used to make gene predictions in the sets of 'complete' as well as 'partial' reference transcripts from *A. thaliana, D. melanogaster*, *M. musculus*, and *S. pombe*. The total numbers of genes predicted in a given set of transcripts could vary depending on the allowed minimum length of predicted gene. We have changed this threshold parameter from 90bp to 480bp with 30bp steps. For each set of predicted genes we computed *Sn* and *Sp* values based on the reference transcript annotation. The dependence of *Sn* value on 1-*Sp* for each gene prediction tool can be plotted as a chart similar to the receiver operating characteristic (ROC) curve (Figure 3.4 and Figure 3.5). The upper right point of each curve is obtained when all predicted genes longer than 90bp are included into consideration. We did not make graphs for ESTscan since we were not able to achieve high enough performance (i.e. for mouse, ESTscan produced Sn=0.53 and Sp=0.54). We believe that the results

could be improved if ESTscan used self-training. However, without such an option we had to select one among available models, e.g. human model for analysis of mouse transcripts.

For the 'complete' reference transcripts, both strand-blind GeneMarkS-T and strand-specific GeneMarkS-T(S) demonstrated significantly better performance, especially in *Sp*, than the two other tools (Figure 3.4). In experiments with 'partial' reference transcripts (Figure 3.5) the Prodigal and TransDecoder were closer in performance to GeneMarkS-T in terms of *Sn*. The values of minimal gene length that delivered the best values of prediction accuracy (*(Sn+Sp)/2*) for GeneMarkS-T, Prodigal, and TransDecoder, were 150bp, 210bp, and 270bp respectively. As expected, we did observe that adding strand-specific information, transition to the gene finders (S) versions, increased the Sp value for each of the three gene finders (Figure 3.4 and Figure 3.5).

In the sets of *M. musculus* and *D. melanogaster* transcripts GeneMarkS-T automatically identified inhomogeneity of the transcript GC composition and grouped the transcripts into three GC content bins. The GC ranges were 31%-46%, 46%-52%, and 52%-76% for the mouse transcripts, and 27%-48%, 48%-51%, and 51%-63% for the fly transcripts. The subsequent self-training done separately in each of the three clusters produced better *Sn* value than in the absence of clustering (Figure 3.1).

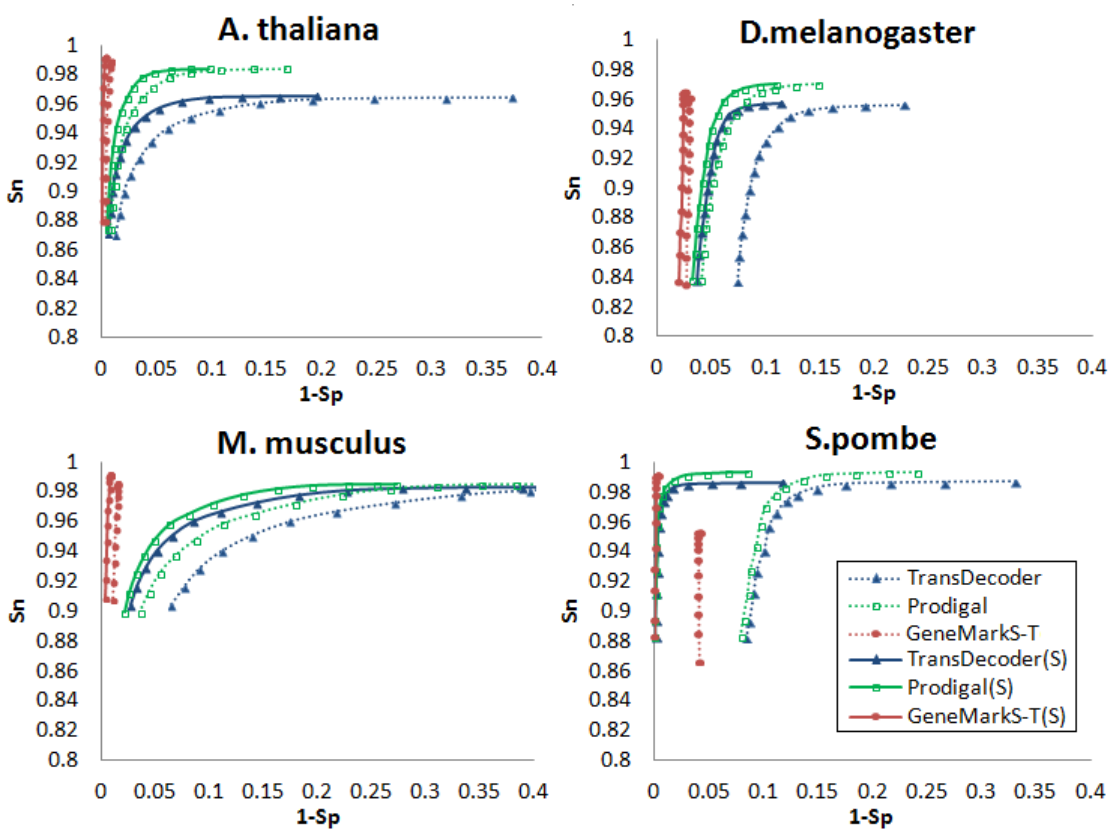

Figure 3.4 The values of gene prediction sensitivity (*Sn*) as functions of gene prediction specificity (1-*Sp*) for TransDecoder, Prodigal, and GeneMarkS-T on the test sets of 'complete' reference transcripts of *A. thaliana, D. melanogaster, M. musculus,* and *S. pombe.*

The three gene prediction methods were applied in both strand blind and strand specific (S) modes. To make the ROC-like curves we generated sets of predicted genes with size controlled by the shortest allowed predicted gene length. This parameter was changing from 90bp to 480bp (with 30bp step); as the minimal allowed length of predicted genes increases the point in the graph moves from higher *Sn* to lower *Sn*.

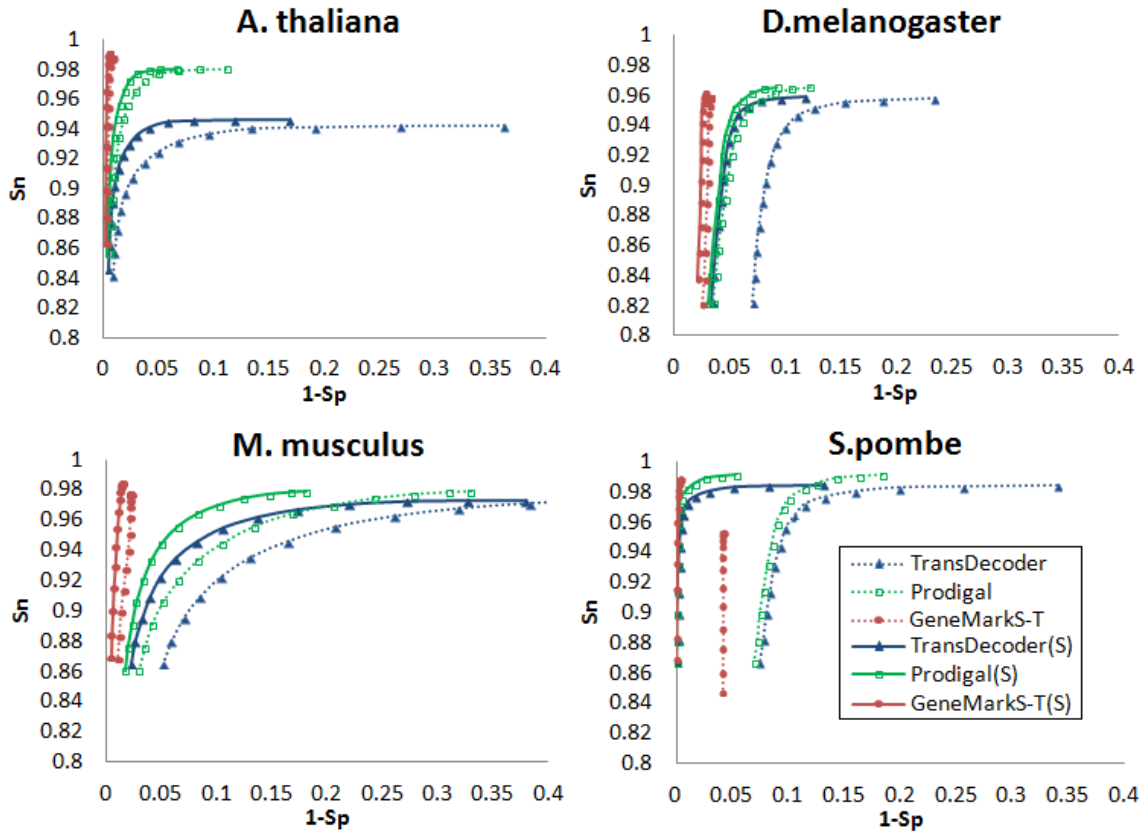

Figure 3.5 Same as in Figure 3.4 for gene prediction in the 'partial' reference transcripts of *A. thaliana, D. melanogaster, M. musculus*, and *S. pombe*.
The 'partial' transcripts were made by trimming 10% of sequences on both 5' and 3' end of the complete transcripts (see a justification of this method in the text). The three gene prediction tools were used in both strand blind and strand specific (S) modes.

Table 3.2 Characteristics of GeneMarkS-T accuracy of gene predictions in reference transcripts of M. musculus and D. melanogaster.
GeneMarkS-T was self-trained with or without dividing transcripts into more G+C homogeneous sets (clusters). The borders of the three clusters were set as 0.31, 0.46, 0.52 and 0.76 for *M. musculus* and 0.27, 0.48, 0.51 and 0.63 for *D. melanogaster*. The mgl (minimal gene length) value was 300bp.

Species	# of clusters	TP	FP	Sn	Sp
D. melanogaster	1	12,007	370	90.7	97.0
	3	12,236	374	92.4	97.0
M. musculus	1	18,346	303	96.9	98.4
	3	18,380	269	97.1	98.6

More than one coding region (longer than 300bp) was initially predicted by GeneMarkS-T in some transcripts (2.5% of *A. thaliana*, 9.4% of *S. pombe*, 6.0% of *D. melanogaster*, and 20.4% of *M. musculus*). Such an outcome is possible if a transcript comes from a genomic locus where splicing mechanism generates alternative isoforms. For instance, protein-coding exons related to one isoform could appear outside the protein coding region related to another isoform (Figure 3.6A). Interestingly, frequent predictions of multiple coding regions were observed in transcripts of *S. pombe*, a species not known for ubiquitous alternative splicing. This observation is likely to be typical for genomes that have short intergenic regions and long UTRs. The long UTRs of *S. pombe* transcripts may overlap adjacent genes situated in the complementary strand (Figure 3.6B). Not surprisingly, a significant gain of accuracy was observed for *S. pombe* after switching to the strand-specific versions of the gene finders (Figure 3.4 and Figure 3.5).

When GeneMarkS-T predicted several coding regions in a single transcript, the prediction with maximum log-odd score was retained. This approach produced 93%

success rate in selecting the 'true' coding region for *A. thaliana*, 74% for *D. melanogaster*, 98% for *M. musculus*, and 62% for *S. pombe*. In *S. pombe*, reduction of the success rate was caused by overlaps between the gene UTRs and genuine coding regions of adjacent genes located in complementary strand (e.g. Figure 3.6B). As mentioned above, use of the strand-specific version of the program was able to eliminate much of the noise.

Further on, we have also studied how gene prediction accuracy depends on the volume of transcripts used in training. We sampled randomly several sets of reference transcripts different in volume to perform self-training and prediction. We observed that if the volume is larger than 600Kb, GeneMarkS-T and Prodigal reached a plateau where the performance is steady and the (Sn+Sp)/2 value is close to 96% for GeneMarkS-T and 94% for Prodigal (Figure 3.7). TransDecoder accuracy had a similar pattern of change with the plateau at 91% reached at the volume of 1Mbp. At 100Kb volume the performance is still high: 90% for GeneMarkS-T and Prodigal, and 80% for TransDecoder. The minimum sequence volume needed for self-training for Prodigal was 20Kb while for GeneMarkS-T the limit is even lower. The reason is that below 50Kb sequence volume, GeneMarkS-T automatically switches to use of heuristic models whose parameters could be determined for a sequence fragment as short as 400bp (Besemer and Borodovsky, 1999).

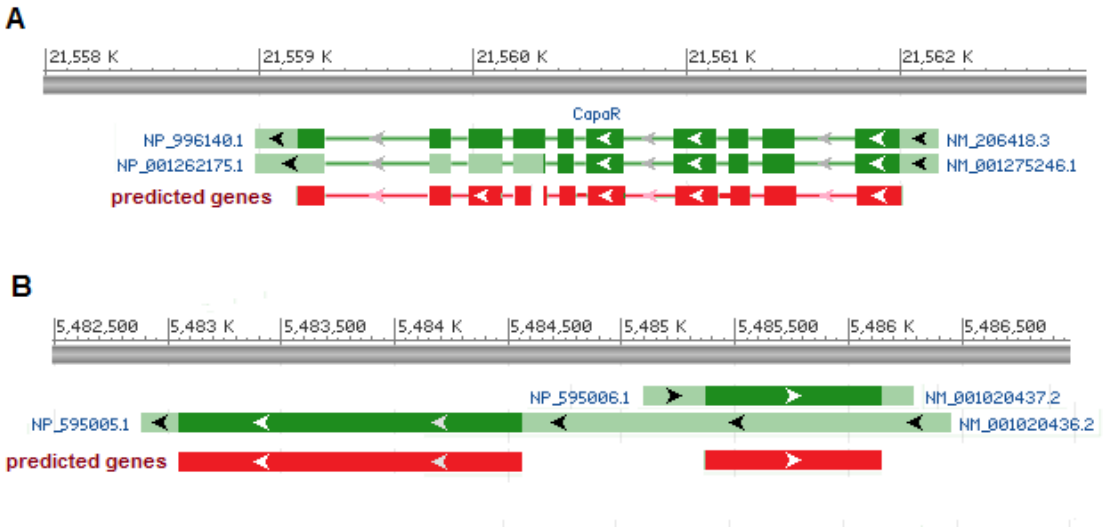

Figure 3.6 Examples of more than one coding regions predicted in a transcript. We show pre-spliced transcripts, with exons defined by annotation shown as wider bars (green color – UTR, dark green - CDS), protein-coding exons predicted by GeneMarkS-T are shown as red bars; genomic sequences are shown as gray bars. (A) Two transcripts were annotated in the same genomic location of the *D. melanogaster* genome (NM_001275246.1 and NM_206418.3). The FP prediction (the downstream gene in the complementary strand) is a part of the coding region in an alternative isoform of the CapaR gene. (B) The 5' UTR of the *S. pombe* transcript NM_001020436.2 overlaps with another transcript NM_001020437.2. GeneMarkS-T made two predictions in NM_001020436.2, in the direct strand (FP) as well as in the complementary strand (TP). The figure was made with the NCBI RefSeq sequence viewer.

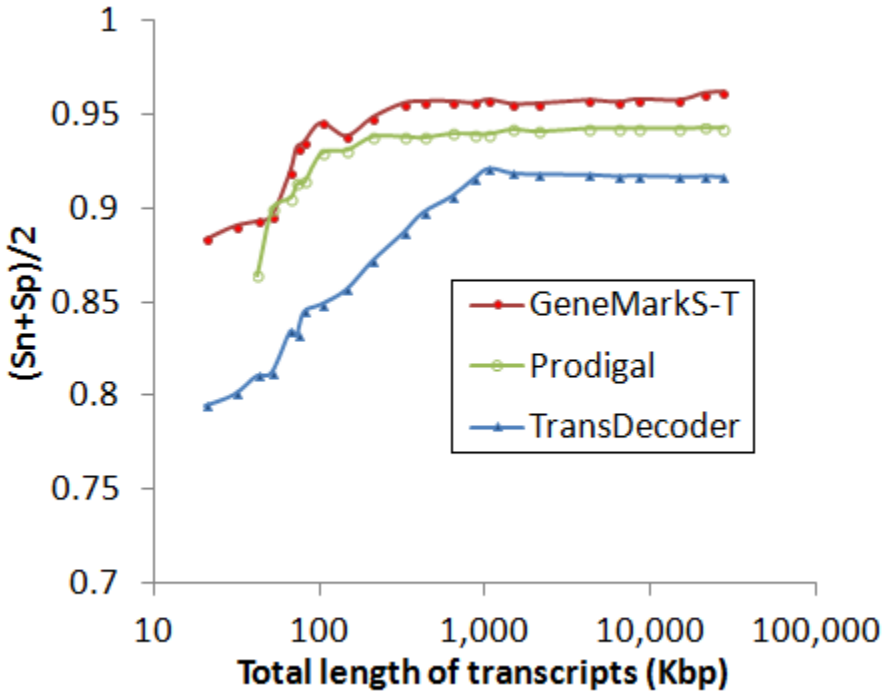

Figure 3.7 Dependence of average *Sn* and *Sp* of the three gene prediction tools trained on the sets of *D. melanogaster* transcripts having different total size (the *X* axis shows the total length, log scale).

The sets of transcripts were sampled randomly from the reference transcripts. A set of transcripts of given size was sampled 50 times to generate the average *Sn* and *Sp* values. Minimal length parameter that achieves best overall average *Sn* and *Sp* was selected for each program (150bp for GeneMarkS-T, 210bp for Prodigal, and 270bp for TransDecoder).

3.3.2 Model Training and Gene Predictions in Reconstructed Transcripts

A comprehensive assessment of the accuracy of several tools of transcript reconstruction from RNA-Seq reads was conducted in RGASP competition (Steijger, et al., 2013). It was shown that assembled transcripts frequently contain errors and only a subset of all transcripts could be fully recovered. The observed average length of assembled transcripts was shorter than that of reference transcripts, particularly the average lengths of *de novo* assemblies made by Oases and Velvet (Figure 3.2A). Would the errors present in transcript assemblies affect the self-training of GeneMarkS-T? To address this question we trained GeneMarkS-T on the sets of *D. melanogaster* transcripts assembled by Cufflinks, Augustus, Velvet, Oases, and Exonerate (Roberts, et al., 2011; Schulz, et al., 2012; Slater and Birney, 2005; Stanke, et al., 2006; Zerbino and Birney, 2008). The model parameters estimated on the five training sets of assembled transcripts were used in GeneMarkS-T to predict protein-coding regions in the reference set of *D. melanogaster* transcripts. The results visualized as graphs of the dependencies of observed *Sn* on the 1-*Sp* values (Figure 3.8) showed almost no difference with the graph depicting Sn dependence over 1-Sp obtained for the case of the parameter training on the reference transcripts. Thus, the GeneMarkS-T training procedure was shown to be robust with respect to transition from "ideal" transcripts to real transcripts.

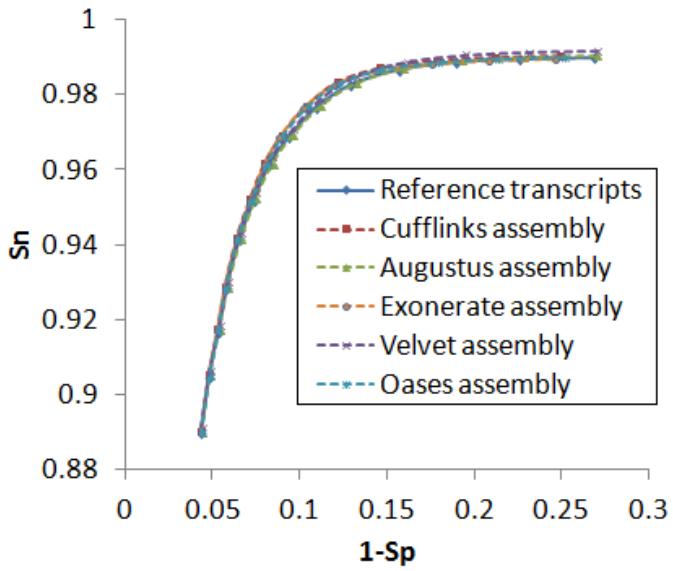

Figure 3.8 Dependence of GeneMarkS-T prediction accuracy on the training set type.
The GeneMarkS-T models were trained either on the sets of *D. melanogaster* transcripts assembled by the five transcript reconstruction tools or on the set of reference transcripts. The predictions were compared with annotations of coding regions in the reference transcripts.

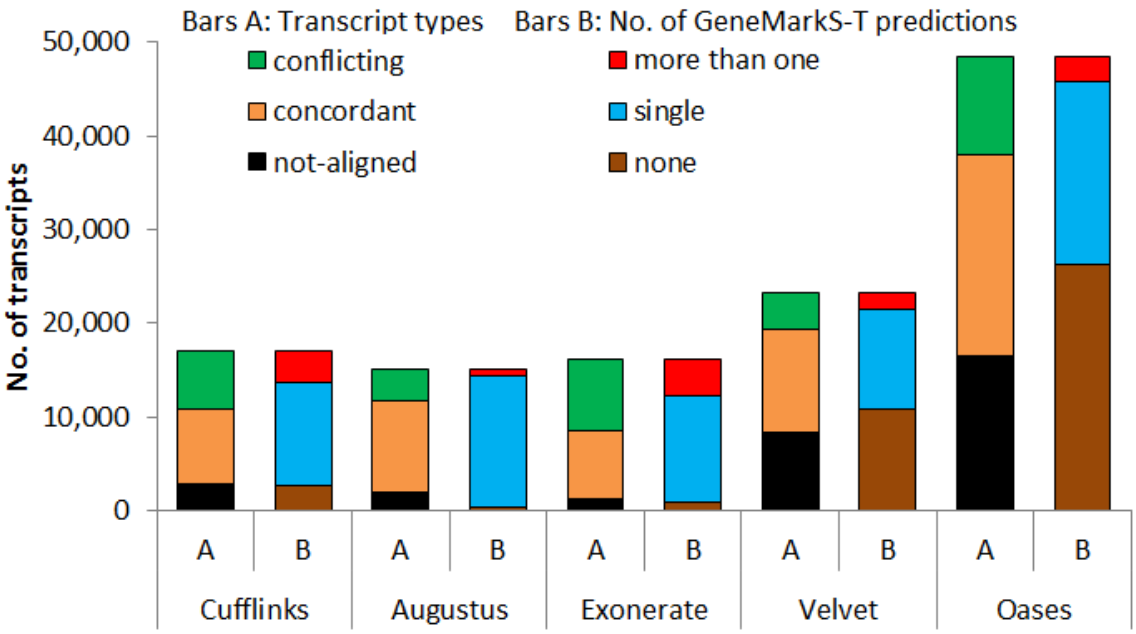

Figure 3.9 Numbers of the three types of assembled transcripts (concordant, conflicting, and not-aligned) as observed in sets of *D. melanogaster* transcripts assembled by the five methods (depicted in bars A).

Numbers of the three types of events: GeneMarkS-T predicting i/ more than one, ii/ single and iii/ none coding regions, in D. melanogaster reference transcripts (depicted in bars B). Predicted coding regions with length less than 300bp were discarded.

Significant fractions of the assembled *D. melanogaster* transcripts were characterized as 'conflicting' (from 17% to 47%, depending on the transcript reconstruction tool, see Figure 3.9, bars labeled 'A'); Cufflinks, Exonerate and Oases produced larger numbers of 'conflicting' transcripts than Augustus and Velvet. Multiple coding regions were predicted more frequently in the 'conflicting' transcripts (Figure 3.10) than in the 'concordant' transcripts (in transcripts assembled by Cufflinks, Exonerate, and Oases the frequency of multiple predictions was higher than in transcripts assembled by Augustus and Velvet).

Reciprocally, in the whole set of assembled transcripts where GeneMarkS-T predicted multiple coding regions, the fraction of 'conflicting' transcripts was high (e.g. 90% for the set of transcripts assembled by Cufflinks). Thus, prediction of multiple coding regions in a transcript indicated a higher chance for the transcript to be in the 'conflicting' category and to carry some discrepancies in the transcript assembly in comparison with the reference transcript. Still, prediction of multiple coding regions in a transcript could indicate that the genomic locus encodes alternative isoforms (as illustrated in Figure 3.6).

Very short coding regions are rare and are rarely predicted. Therefore, if an assembled transcript (complete or incomplete) is short it is very likely that no gene will be predicted. Indeed, the gene finding tools used in this study did not predict genes in many transcripts assembled by the *de novo* methods (Velvet and Oases, Figure 3.9), notably, many of these transcripts were too short (Figure 3.2A).

We observed significantly larger frequencies of prediction of single coding regions in the 'concordant' assemblies than in 'conflicting' assemblies (see Figure 3.10 made for GeneMarkS-T predictions). The distribution of numbers of predictions of single coding regions for the two other gene prediction tools shows similar distribution patterns (Table 3.3). Thus, we can argue that all the three tools predict single coding regions in 'concordant' assemblies with much higher frequencies than in 'conflicting' ones.

Table 3.3 Numbers of the three types of events:
Predicting i/ more than one, ii/ single and iii/ none coding regions by GeneMarkS-T, Prodigal, and TransDecoder in *D. melanogaster* transcripts of concordant type reconstructed from RNA-Seq reads by Cufflinks, Augustu, Exonerate, Velvet, and Oases. The *mgl* value was 300bp.

Assembly method	# of concordant transcripts	Prediction tool	# of predicted coding regions		
			>1	1	0
Cufflinks	7,886	GeneMarkS-T	236	7,220	430
		Prodigal	184	7,188	514
		TransDecoder	483	6,828	575
Augustus	9,834	GeneMarkS-T	191	9,446	197
		Prodigal	139	9,431	264
		TransDecoder	502	9,017	315
Exonerate	7,375	GeneMarkS-T	231	6,971	173
		Prodigal	189	6,985	201
		TransDecoder	537	6,612	226
Velvet	11,032	GeneMarkS-T	135	7,320	3,577
		Prodigal	109	7,244	3,679
		TransDecoder	324	6,967	3,741
Oases	21,409	GeneMarkS-T	306	13,830	7,273
		Prodigal	297	13,653	7,459
		TransDecoder	696	13,221	7,492

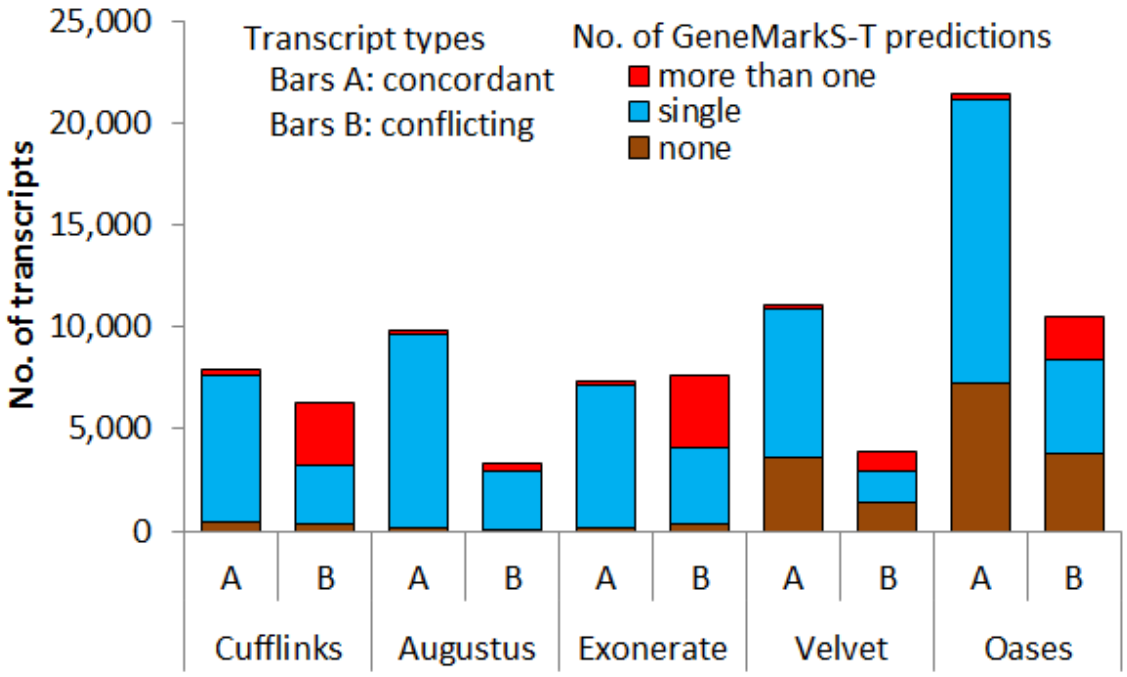

Figure 3.10 Numbers of the three types of events: GeneMarkS-T predicting i/ more than one, ii/ single and iii/ none coding regions, in *D. melanogaster* concordant (bars A) and conflicting transcripts (bars B).
The minimal gene length (*mgl*) value was 300bp. Events of prediction of multiple coding regions were registered prior to selecting 'reported' predictions with highest log-odd score.

From the sets of 'concordant' assemblies generated by the five tools we removed those transcripts where GeneMarkS-T predicted multiple protein-coding regions. Annotation of protein-coding regions in the assemblies selected for this test was done by transfer of the reference transcript annotation. Next, the three gene finding tools were run to produce gene predictions that were compared with annotations. In all the five test sets, GeneMarkS-T generated the largest number of TPs and the fewest number of FPs (Table 3.4).

Table 3.4 Numbers of protein-coding regions predicted correctly (TP) and incorrectly (FP) by GeneMarkS-T, Prodigal, and TransDecoder in a set of *D. melanogaster* 'concordant' assembled transcripts.
Predictions shorter than the tool-specific minimum length (150bp for GeneMarkS-T, 210bp for Prodigal, and 270bp for TransDecoder) were filtered out.

Transcript built by	No. of transcripts	GeneMarkS-T		Prodigal		TransDecoder	
		TP	FP	TP	FP	TP	FP
Cufflinks	7,222	7,162	60	7,098	232	7,046	432
Augustus	9,444	9,423	21	9,383	246	9,332	480
Exonerate	6,971	6,953	18	6,940	190	6,915	454
Velvet	7,344	7,146	198	7,096	312	7,030	429
Oases	13,869	13,769	100	13,659	347	13,598	582

3.3.3 Translation Initiation Site Prediction

To assess the accuracy of GeneMarkS-T, Prodigal, and TransDecoder in TIS prediction we used 1,392 reference transcripts of *M. musculus* (with annotated coding regions longer than 300bp). The TIS annotation in these transcripts was validated by Ribo-seq experiments (see section 3.2.2). GeneMarkS-T was run in three modes: i/ with default settings; ii/ with search for the Kozak motif switched off; and iii/ with making

predictions of complete CDSs only (translation initiation start and translation stop within the transcript).

GeneMarkS-T demonstrated better performance in TIS identification than two other tools (Table 3.5). All three programs revealed a tendency to extend the 5' end of the coding region beyond the 5' end of the transcript. Notably, TransDecoder adopts the "longest-ORF" rule and selects the 5'-most AUG as the translation initiation site. In comparison, GeneMarkS-T had the largest fraction of TIS predictions located downstream from the 5'-most AUGs. Although prohibiting predictions of incomplete coding regions would boost the TIS identification accuracy to 95%, use of this option is limited to transcripts that are known to be 5' end complete. Use of the Kozak motif improved *Sn* of predicting TIS by about 10% (Table 3.5). Nonetheless, even the highest *Sn* value achieved by GeneMarkS-T was smaller than 70%.

Several ribosome profiling studies (Fritsch, et al., 2012; Ingolia, et al., 2011; Lee, et al., 2012) raised concerns about frequent presence of alternative TIS's both upstream and downstream of annotated TIS's confirmed by Ribo-seq experiments. However, recent publication (Gerashchenko and Gladyshev, 2015) indicated that findings of alternative TIS in many cases are likely to be artifacts; therefore, the confidence in the Ribo-seq experimental validation of annotated TISs remains high.

Table 3.5 Results of assessment of gene prediction accuracy of GeneMarkS-T, Prodigal, and TransDecoder on the set of 1,392 mouse transcripts with experimentally verified translation initiation sites (coding regions length >300bp). The columns show (from left to right) the number of genes i/ with 3' ends correctly identified and its fraction (%) in the whole set of transcripts; ii/ exactly predicted (both 5' and 3' ends correctly identified) and its fraction (%) among genes with correctly predicted 3' ends; iii/ not matching annotation in 3' end (false positives); iv/ predicted shorter than annotated; v/ predicted longer than annotated, with number of predicted genes with 5' end beyond the 5' border of actual transcript sequence (incomplete predictions) shown in parentheses. The results are also shown for GeneMarkS-T runs 1/ without model for the Kozak motif; 2/ with requirement to predict 5' complete genes; 3/ analyzing each transcript independently with use of only one iteration; parameters of heuristic models for each transcript were selected as functions of the given transcript G+C content (simulation of a run on meta-transcriptome).

	Exact 3' end	Exact 5' and 3' ends	#FP	#Shorter	#Longer
Prodigal	1,193 (85.7%)	612 (51.3%)	351	9	572 (571)
TransDecoder	1,193 (85.7%)	623 (52.2%)	428	0	570 (568)
GeneMarkS-T	**1,197** (**86.0%**)	**821** (**68.6%**)	**195**	43	333 (333)
GeneMarkS-T[1]	1,196 (85.9%)	694 (58.0%)	196	51	451 (450)
GeneMarkS-T[2]	1,194 (85.8%)	1,134 (95.0%)	197	59	1 (0)
GeneMarkS-T[3]	1,147 (82.4%)	630 (54.9%)	321	259	258 (204)

3.4 Discussion

Here we summarize our observations on the performance of GeneMarkS-T, Prodigal, and TransDecoder. As we saw in the Results section, comparison of the three tools runs on the sets of complete and partial reference transcripts have demonstrated higher performance of GeneMarkS-T in comparison with the other two gene finders (Figure 3.4 and Figure 3.5). Notably, the GeneMarkS-T "minimal gene length" threshold of 150bp was the lowest among the three. This setting indicated that GeneMarkS-T works more accurately in the short gene range. Prodigal (TransDecoder) had to filter out

predictions shorter that 210bp (270bp) that contained more false positives than true positives, thus decreasing the (Sn+Sp)/2 value.

To model 'partial' transcripts we used statistics on the types of 'partial' structures (Figure 3.2A) observed in experiments on transcript reconstruction (Steijger, et al., 2013). We observed that the *Sn* value of GeneMarkS-T predictions in partial transcripts dropped slightly (<1%), however, the performance was still better in comparison with the two other tools (Figure 3.5).

Use of the strand-specific versions of the three tools increased prediction accuracy in the test sets for all the four species (Figure 3.4 and Figure 3.5). The largest gain was observed in tests on *S. pombe*, which, among the four species, has the highest density of coding regions and, therefore, more frequent occurrences of events when a UTR overlaps a part of adjacent gene in complementary strand causing additional gene prediction in a single transcript (Figure 3.3b). The strand-specific versions of the tools have to be used for transcripts assembled from reads generated by the strand-specific RNA-Seq technique.

We observed that the GeneMarkS-T gene prediction accuracy in mouse and fly *reference* transcripts was improved by clustering transcripts with similar GC content (Table 3.2). We have also shown that accuracy of GeneMarkS was not affected by a decrease of the size of the transcript set used for self-training, even down to 100Kb (Figure 3.7).

We have shown that training of GeneMarkS-T was robust with respect to transition from training on reference transcripts to training on transcripts *assembled* from

RNA-Seq reads (Figure 3.8). Notably, the training quality was not affected by large numbers of short incomplete transcripts generated by Velvet and Oases (Figure 3.2A, Figure 3.9). The short fragments were effectively removed from training, since genes were not predicted in the short sequences.

We also observed that multiple gene predictions in a single transcript were much more frequent in *assembled* than in *reference* transcripts (Table 3.3). Notably, the frequency of multiple predictions was lower in transcripts reconstructed by Augustus which attempted to preserve continuous coding potential upon assembling RNA-Seq reads.

Assessment of accuracy of gene prediction in assembled transcripts was challenging due to the presence of assembly errors. Study of the assembly errors that occur upon application of existing transcript reconstruction tools is a special topic. A comprehensive comparative assessment of the methods of transcript reconstruction from RNA-Seq reads was made in the RGASP competition (Steijger, et al., 2013) that used sets of RNA-Seq reads generated for *Homo sapiens*, *Caenorhabditis elegans*, and *Drosophila melanogaster*. The average accuracy of transcript reconstruction was shown to be genome-specific, e.g. the *C. elegans* transcripts were reconstructed most accurately on average. Nonetheless, only 48% of the *C. elegans* transcripts were assembled correctly.

Assessment of accuracy of gene prediction was made for *D. melanogaster* transcripts assembled from RNA-Seq reads by the five tools, Cufflinks (Roberts, et al., 2011), Augustus (Stanke, et al., 2006), Velvet (Zerbino and Birney, 2008), Oases

(Schulz, et al., 2012), and Exonerate (Slater and Birney, 2005) that were taking part in the RGASP competition (Steijger, et al., 2013). Each of the five sets of transcripts was divided into 'concordant', 'conflicting', or 'not-aligned' subsets (Figure 3.3).

We observed that multiple genes were predicted in 'conflicting' transcripts much more frequently than in 'concordant' transcripts. On the other hand a question could be asked, what is the chance that a transcript where multiple genes were predicted belongs to 'conflicting' or 'not-aligned' category? The answer to the question is species and tool specific, e.g. for Cufflinks it is 93% (Figure 3.10). Arguably, prediction of multiple genes in a single transcript is an indicator of erroneous assembly.

Erroneous assembly could produce a gap or insertion in the transcript, a chimeric fusion with another transcript, etc. A frequent consequence of an incorrect assembly is a frameshift in a protein-coding region. Earlier we developed computational tools for finding frameshifts in continuous protein coding regions such as GeneTack (Antonov, Baranov, et al., 2013; Antonov and Borodovsky, 2010; Antonov, Coakley, et al., 2013). Integration of GeneMarkS-T with a GeneTack type tool would make a new tool able to provide deeper insight into organization of coding regions in assembled transcripts. However, besides this extension being beyond a scope of the current work, the usefulness of such approach may be limited since a presence of several gene predictions in a transcript (presumably caused by a frameshift) is already a good indicator of incorrect assembly. The best way to deal with this problem is to check and fix the assembly error(s) rather than to proceed with conceptual translation based on predicted frameshift location(s).

Assessment of gene prediction accuracy in assembled transcripts should be therefore limited to the transcripts where protein coding regions were not disrupted, i.e. the 'concordant' transcripts. Runs of GeneMarkS-T, Prodigal, and TransDecoder on the five sets of 'concordant' transcripts of *D. melanogaster* demonstrated that GeneMarkS-T delivered more accurate predictions (Table 3.4).

We used mouse transcripts with the translation initiation site annotation validated by Ribo-seq experiments to assess accuracy of the TIS prediction. Even though GeneMarkS-T demonstrated better accuracy in comparison with the other two tools (Table 3.5), the accurate TIS prediction remains a challenge as the accuracy achieved by GeneMarkS-T is still below 70%. Efforts aiming to further improvement of the TIS prediction accuracy will benefit from availability of a larger set of validated TIS. The novel Ribo-seq technique is likely to be instrumental in generating such a larger set.

Finally, what could explain better accuracy of GeneMarkS-T in gene prediction in eukaryotic transcripts in comparison with the other two gene finders? Notably, Prodigal is using the iterative training approach similar to GeneMarkS and GeneMarkS-T. TransDecoder is using oligomer statistics to identify protein coding region similarly to Prodigal and GeneMarkS-T. First, we assume that the higher accuracy manifests the ability of the algorithm to scale up to new types of genomic sequences, e.g. eukaryotic transcripts or short sequences of metagenomes. The use of hidden semi-Markov model provides a necessary degree of flexibility to adjust to the new types of organization of coding regions that are different from organization of genes in complete genomes. Perhaps, the structure of the Prodigal, the algorithm that is using multiple smart heuristics, is more difficult to scale up to another type of application. On the other hand,

the structure of the TransDecoder algorithm is relatively simple and rather crude to take into account features of gene organization in complete and incomplete eukaryotic transcripts. Second, GeneMarkS-T has the built-in option to cluster a full set of transcript sequences into more homogeneous sets; selection of cluster-specific model for a given transcript increases prediction accuracy.

GeneMarkS-T can also be applied to meta-transcriptomic sequences for gene discovery. Like in eukaryotes, many studies have adopted transcriptome sequencing to reveal expressed genes and active pathways in both individual microbes (Martin, et al., 2010; Passalacqua, et al., 2009; Sharma, et al., 2010; Wurtzel, et al., 2010) and complex microbial communities (Booijink, et al., 2010; Frias-Lopez, et al., 2008; Turnbaugh, et al., 2010; Turner, et al., 2013). In typical RNA-Seq applications, sequencing reads can be assembled into longer stretches by reference-based methods or by *de novo* assembly methods (evaluated in (Sekhar, et al., 2011)). Correctly assembled transcripts from individual microorganisms resemble fragmented genomic sequences, thus can be analyzed using tools and algorithms ready for microbiome genomes. Meta-transcriptomic sequences, in contrast, have more complex sequence content with anonymous origin, similar to metagenomic sequences. Therefore, special ways of parameter estimation should be used for meta-transcriptomic sequence analysis.

Methods that have been established for metagenomic gene prediction can be applied to meta-transcriptomic sequence analysis. These algorithms include heuristic method for parameter estimation (Zhu, et al., 2010), frameshift identification in reads containing sequencing errors (Tang, et al., 2013), motif-finding approaches, and partial gene identification. Additional features unique to meta-transcriptomic data can also be

leveraged. One assembled sequence represents a full or partial stretch of a transcript; it can contain zero, one, or more than one coding gene (transcribed together in a functioning unit known as an "operons"). Genes in an operon are on the same strand, and often overlap with each other or have short intergenic regions. The strand information and length distribution of the intergenic regions of genes in an operon can be modeled accordingly in the HSMM. In addition, since genes encoded in mRNAs are in the positive strand, strand-specific RNA-Seq protocol (evaluated in (Levin, et al., 2010)) preserves the strand information of the sequence reads. Therefore, eliminating gene prediction in the wrong strand would facilitate gene prediction as well as anti-sense transcript identification. A special option of GeneMarkS-T allows the prediction of more than one protein coding genes in one transcript with the use of heuristic models. This option also searches for the RBS in the upstream sequences instead of the Kozak motif, which can improve the prediction of the gene boundary in prokaryotic transcripts.

While the above strategies are intuitive and straight-forward to implement, the self-training method used in eukaryotic transcripts may also be applicable to meta-transcriptomic data. For transcripts from an inhomogeneous genome, sequences are clustered into GC bins before training of the native model. Meta-transcripts are also inhomogeneous as they may have different taxonomic assignments. Clustering Meta-transcripts of the same GC or similar sequence signatures may find homogenous sequences in terms of their functional assignments. Models trained on the clusters could predict genes additional to heuristic models, adding genes that are "native" to a certain function of the whole transcriptome. However, the composition and relative abundance of

different organisms in the microbial community should be studied carefully to determine the level of homogeneity thus the usefulness of self-training.

3.5 Software Availability

The GeneMarkS-T software is freely available for academic research and can be downloaded from http://topaz.gatech.edu/GeneMark/license_download.cgi.

CHAPTER 4
FRAMESHIFT PREDICTION IN METAGENOMIC SEQUENCES

Abstract

Frameshift (FS) prediction is important for analysis and biological interpretation of metagenomic sequences. Since a genomic context of a short metagenomic sequence is rarely known, there is not enough data available to estimate parameters of species-specific statistical models of protein-coding and non-coding regions. The challenge of *ab initio* FS detection is, therefore, twofold: (i) to find a way to infer necessary model parameters and (ii) to identify positions of frameshifts (if any). Here we describe a new tool, MetaGeneTack, which uses a heuristic method to estimate parameters of sequence models used in the FS detection algorithm. It is shown on multiple test sets that the MetaGeneTack FS detection performance is comparable or better than the one of earlier developed program FragGeneScan.

4.1 Introduction

Metagenomic sequences are obtained from environmental microbial communities. The short reads sequenced using next-generation sequencing technology are then processed and assembled to sequence contigs. Depending on the metagenomic sequence data structure, gene prediction is performed on sequence contigs, unassembled reads, or a mixture of them (Kunin, et al., 2008). The performance of conventional tools of gene prediction and annotation are impaired due to insertions and deletions in coding regions of the short sequences (Hoff, 2009). Error rates depend on various factors including species complexity, genome abundance, the sequencing method and assembly strategies (Luo, Tsementzi, Kyrpides and Konstantinidis, 2012). Since an average metagenomic contig length is of the order of several hundred to several thousand nucleotides, there is not enough sequence contexts to estimate parameters of statistical models for protein-coding and non-coding regions. On the other hand, comparative genomics based tools (looking for FSs interrupting evolutionary conserved regions) rely entirely on existing databases; those tools would miss novel genes and genes that have low similarity with known genes.

Previously we have developed an algorithm and software program GeneTack (Antonov and Borodovsky, 2010), an *ab initio* tool for finding frameshifts (FSs) in prokaryotic genomes. Since GeneTack requires a species-specific statistical model, it cannot work with sequences shorter than several hundred kilobases, the length necessary for self-training of GeneMarkS (Besemer, et al., 2001), gene finder used together with GeneTack. Here we introduce a new *ab initio* FS finder designed for metagenomic

sequences, MetaGeneTack, which uses heuristic models (Besemer and Borodovsky, 1999) to infer model parameters suitable for analysis of a short sequence (e.g., 400nt). A recently developed *ab initio* gene finder, FragGeneScan (Rho, et al., 2010) is also able to detect positions of FSs in short sequences by adding insertion/deletion states in the architecture of the hidden Markov model. In test on sequences from 18 prokaryotic species we have shown that MetaGeneTack reaches higher accuracy in FS detection than FragGeneScan.

4.2 Materials and Methods

The idea of the heuristic method for building models of protein-coding regions is that frequencies of oligonucleotides, if cannot be derived directly due to insufficient sequence length, can be inferred as functions of the sequence GC content. Thus, we could reconstruct the oligonucleotide frequencies as soon as we have an estimate of the GC content of genome the short sequence originated from. MetaGeneTack uses the fifth-order polynomial approximations of dependences of hexamer frequencies on genome GC content derived from data on 582 annotated prokaryotic genomes (the details for the model parameter derivation can be found in (Zhu, et al., 2010).

MetaGeneMark (Zhu, et al., 2010), a tool designed for *ab initio* gene prediction in metagenomic sequences is used for initial prediction of protein-coding genes. The GeneTack algorithm works with coding regions located in the positive strand; thus the input sequences are split into fragments with predicted genes located in the same strand, grouped by GC content. The grouped fragments are then used as input for GeneTack with the corresponding heuristic model (of bacterial or archaeal type of the same GC content). The type of the model for a given fragment is defined by MetaGeneMark which labels

the predicted genes as archaeal or bacterial. FS predictions are combined with gene prediction result from MetaGeneMark output to indicate genes with or without FSs.

To reduce the number of false positives, MetaGeneTack contains three post-processing filters applied to the initial FS predictions. A frameshift would cause two overlapping ORFs with high coding potential. The downstream ORF predicted as a gene should not possess a functional RBS site. Therefore, if a gene predicted in the downstream ORF has high RBS score, the prediction is filtered out (Filter I). In high-GC genomes, true FS would be separated by a long distance from a stop codon terminating the upstream overlapping ORF. A predicted FS situated on a short distance from the stop codon terminating the upstream ORF ($D(\theta) < 0.8\theta - 40$), with θ designating GC content in percentage scale is filtered out (Filter II). A FS predicted too close (<50nt) from a border of a putative frame-shifted gene or from 3' or 5' end of the sequence fragment is discarded (Filter III). Filter II and III are applied only to fragments with high GC content ($\theta > 50$) or low GC content ($\theta \leq 50$), respectively. As a training set for assignment of the filters' parameters we used genomic sequences of *E. coli.*

Metagenomic sequences are usually sequenced using next generation sequencing platforms such as Roche 454 and Illumina or traditional Sanger sequencing. 454 sequencing platforms produce reads of ~450bp; errors are usually indels in homopolymer regions. Illumina platforms generate sequences of length ~100bp with almost no FS errors. Sanger sequencing produces reads that may contain both types of errors and the read length is ~900bp. Before gene calling is performed, metagenomic pipelines usually consist of quality control methods to reduce errors on raw reads (e.g. trimming the error-prone 3'ends). In sequence contigs, the per-base error rate can be reduced from 0.5% in

raw reads to as low as 0.005% and errors affect ~3% to 4.5% of genes (Luo, Tsementzi, Kyrpides, Read, et al., 2012). To evaluate the accuracy of FS detection, we used 18 prokaryotic genomes with GC content ranging from 28% to 75% (Table 4.1). These genomic sequences were cut into 400nt, 600nt and 800nt fragments. Selection of the 400nt as the minimum fragment length is in agreement with the conventional practice where fragments shorter than 400nt are used for detecting nucleotide polymorphisms and short functional motifs (Wooley, et al., 2010). We selected 2,000 fragments of each length from every genome and 5%, 10%, and 20% of all fragments were simulate with a FS (dividing by the corresponding fragment length would provide the per-base error rate, ranging from 0.0065% to 0.05%). In the simulation, it was made sure that the selected fragments contained a long stretch of coding regions (>200nt) and one nucleotide was inserted at a random location in the coding region at a distance of at least 50nt from its boundary. If a FS was predicted in the 20nt vicinity of the true FS position, it was reported as a true positive, otherwise as a false positive. The same test for each genome was done when deletion FSs were simulated.

Table 4.1 Frameshift prediction accuracy for 400nt fragments from 18 prokaryotic genomes (with 20% containing FSs).

"Avg" denotes the average of sensitivity (*Sn*) and specificity (*Sp*). In general, MetaGeneTack reaches higher accuracy for bacteria than archaea (indicated by stars in the table). Interestingly, greater sensitivity of MetaGeneTack in frameshift detection is observed in high GC and low GC genomes than middle GC genomes. Archaeal genomes are marked with a star.

			FragGeneScan			**MetaGeneTack**		
	ID	GC%	*Sn*	*Sp*	*Avg*	*Sn*	*Sp*	*Avg*
Methanosphaera stadtmanae *	NC_007681	28	87.3	62	74.6	74.9	83.8	79.3
Campylobacter jejuni	NC_002163	31	88.6	50.6	69.6	84.3	67.3	75.8
Staphylococcus aureus Mu50	NC_002758	33	88.6	51.9	70.2	76.6	83.1	79.8
Picrophilus torridus DSM 9790 *	NC_005877	36	52	24.5	38.3	74.4	70.8	72.6
Streptococcus pyogenes M1 GAS	NC_002737	39	87.6	47.6	67.6	70.3	76.9	73.6
Pasteurella multocida	NC_002663	40	85.3	57	71.1	70.8	75.5	73.1
Bacillus subtilis	NC_000964	44	78.3	39.6	58.9	64.3	71.9	68.1
Thermotoga maritima	NC_000853	46	60.8	28.7	44.7	66.8	61.5	64.1
Archaeoglobus fulgidus *	NC_000917	49	63.9	24.9	44.4	80.4	58.8	69.6
Escherichia coli K12	NC_000913	51	83.6	42.6	63.1	76.8	71.1	73.9
Pyrobaculum aerophilum *	NC_003364	51	65	27.8	46.4	60.1	55.1	57.6
Salmonella typhimurium LT2	NC_003197	52	85.6	43.9	64.8	75.4	70.7	73
Thermococcus kodakaraensis	NC_006624	52	69.1	27.8	48.5	79.4	59.7	69.5
Methanopyrus kandleri *	NC_003551	61	81.3	36.1	58.7	68.3	60.4	64.3
Caulobacter crescentus	NC_002696	67	94.6	59.7	77.2	83.8	73.3	78.5
Ralstonia solanacearum	NC_003296	67	94.3	48.8	71.5	86.6	70.7	78.7
Clavibacter michiganensis	NC_010407	73	95	47.3	71.2	83.9	70.9	77.4
Anaeromyxobacter dehalogenans	NC_007760	75	96.4	57.5	76.9	87.1	82.4	84.8
Average			81	43.2	62.1	75.8	70.2	73

4.3 Results

Using *A* to denote the number of all FS predictions, *T* to denote the number of predicted true positives, and *S* to denote the number of simulated FSs, we calculated sensitivity, *Sn=T/S* and specificity *Sp=T/A*. Accuracy of MetaGeneTack was compared with accuracy of FragGeneScan[8]. FragGeneScan requires users to select a sequencing method presumably used for obtaining the input sequence along with indication of approximate sequencing error rate. We chose Sanger sequencing with 0.5% as the error rate matched the one cited in (Luo, Tsementzi, Kyrpides, Read, et al., 2012), and it yielded the best results of FragGeneScan among all available options. The average *Sn* and *Sp* values are shown in Table 4.2. To give an example of genome-specific values of *Sn* and *Sp*, we provide Table 4.1 for the set of 400nt fragments with 20% containing FSs. Results are averaged between sets of fragments with insertions and deletions (see also Figure 4.1).

In terms of *(Sn+Sp)/2*, MetaGeneTack performed better than FragGeneScan by 7% to 12%. For FragGeneScan, the values of *Sn* and *Sp* differed by 55 percentage points while for MetaGeneTack this gap was much smaller. The differences were likely due to different methods of derivation of sequence models and differences in HMM architectures.

---

[8] version 1.15, downloaded from http://omics.informatics.indiana.edu/FragGeneScan/

Table 4.2 FS detection accuracy of FragGeneScan and MetaGeneTack for short fragments from 18 prokaryotic genomes.

Values are averaged among genomes and then averaged between insertion and deletion FS sets (see Table 4.1 for details).

Fragment length	Fragments with FSs	FragGeneScan			MetaGeneTack		
		Sn	*Sp*	*Avg*	*Sn*	*Sp*	*Avg*
400nt	5%	79.6	15.8	47.7	74.4	38.3	56.4
	10%	80.5	27.3	53.9	75.3	54.5	64.9
	20%	81	43.2	62.1	75.8	70.2	73
600nt	5%	81.2	11.7	46.4	79.9	27.7	53.8
	10%	81.8	21.2	51.5	79.9	43.1	61.5
	20%	81.9	35.1	58.5	80.1	61.7	70.9
800nt	5%	81.9	9.1	45.5	81.7	21.7	51.7
	10%	82.6	16.9	49.7	81.2	35	58.1
	20%	82.8	29.4	56.1	81.5	51.9	66.7

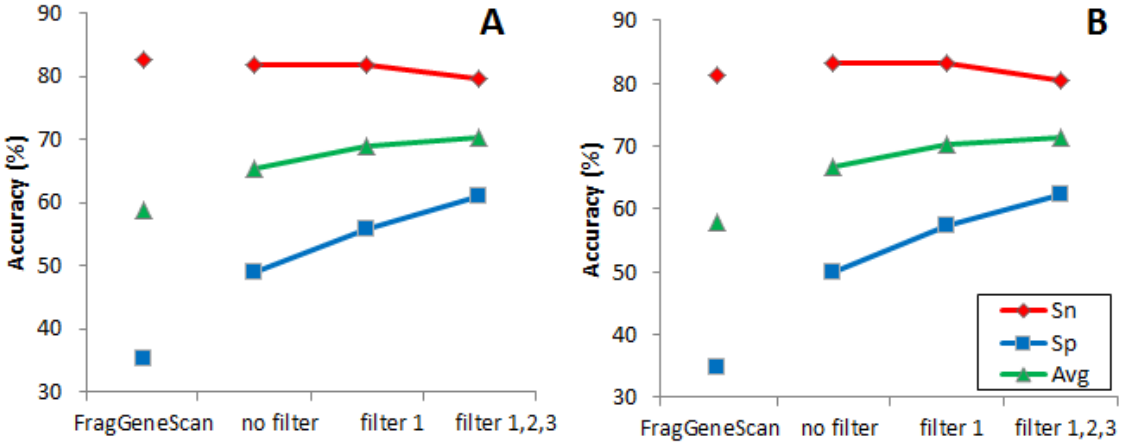

Figure 4.1 Performance of MetaGeneTack with different combinations of filters as well as performance of FragGeneScan (the leftmost columns) using the 600 nt sequences with 20% having simulated FSs as the test set.
The predicted frameshift is reported as true positive if it is located within 20nt from the true simulated frameshift position, (A) for fragments with insertions, (B) for fragments with deletions. Values are averaged among 18 genomes

To assess how effective the filters were we evaluated MetaGeneTack's performance produced with various combinations of filters and compared with performance of FragGeneScan on insertion FS (Figure 4.1A) and deletion FS (Figure 4.1B). Here we used 600nt-long sequences with 20% fragments containing FSs as a test set. Without filters, the *Sn* of MetaGeneTack was close to FragGeneScan while the *Sp* was more than 10% higher in both cases. With the filters, the average *Sn* and *Sp* of MetaGeneTack increased by ~5 percentage points. Similar results were observed when a prediction was reported as a true positive if located within 10nt from the simulated FS. The distribution of the difference between the predicted FS position and the real FS position is shown in Figure 4.2. The standard deviation is 10.3 and 12.6 for MetaGeneTack and FragGeneScan, respectively.

The performance of both programs on error-free sequences was also analyzed on fragments of various lengths. In each genome we used 1000 fragments without simulated FSs. On average, FragGeneScan produced twice as much false positive prediction as MetaGeneTack (Table 4.3).

Table 4.3 Frameshift predictions in 18,000 "frameshift-free" sequences (1,000 for each genome).

	FragGeneScan		MetaGeneTack	
Fragment Length	No. of predicted FSs	Error per nt	No. of predicted FSs	Error per nt
400nt	4,309	0.06%	1,942	0.03%
600nt	6,459	0.09%	3,433	0.05%
800nt	8,696	0.12%	4,978	0.07%

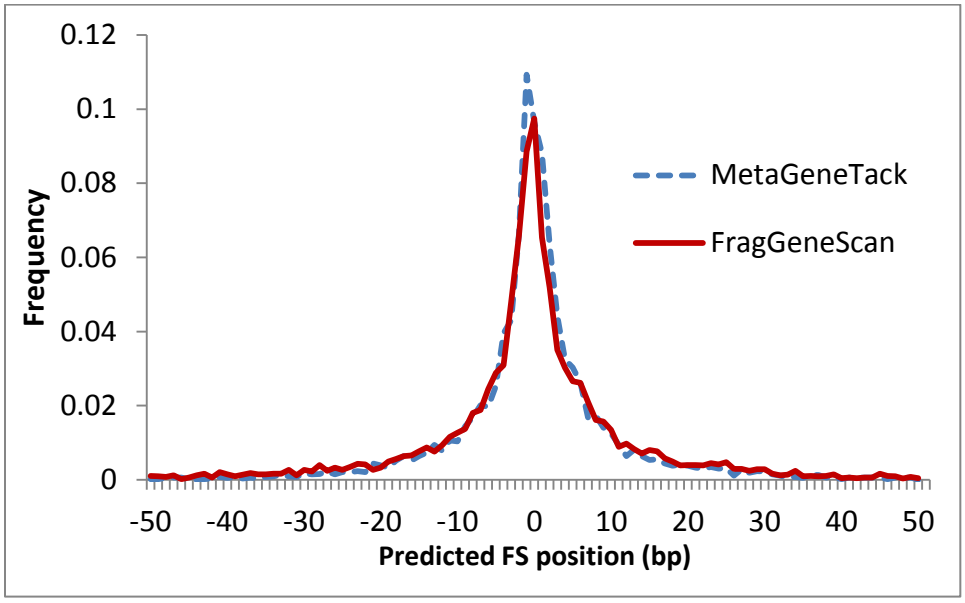

Figure 4.2 Distributions of the distance between predicted FS positions and true FS positions for 400nt, 600nt, and 800nt fragments with simulated FSs.
Deviation longer than 50nt is not shown. The standard deviation is 10.3 and 12.6 for MetaGeneTack and FragGeneScan, respectively.

4.4 Conclusion

The new software program, MetaGeneTack, addresses the challenging question of how to predict FSs in metagenomic sequences without any extrinsic knowledge. An advantage of *ab initio* approach is the ability to detect FSs in genes of orphan proteins that do not have known homologs. We have shown that the accuracy of MetaGeneTack is higher than the accuracy of the *ab initio* gene prediction tool FragGeneScan. Most of the FSs predicted by MetaGeneTack are supposed to result from sequencing or assembly errors; like GeneTack, the program is also able to detect programmed FSs and FSs because of mutations. MetaGeneTack could be integrated into pipelines of metagenomic sequence annotation.

APPENDIX

SUPPLEMENTARY DATA

Supplementary Table 1 The test set of 115 bacterial and 30 archaeal RefSeq genomes.

The number of false negative predictions (FNs) of COG-supported genes, and the average number of false positive predictions (FPs) in 10 simulated 1Mbp non-coding random sequences by GeneMarkS, Glimmer, Prodigal, and GeneMarkS-2 are shown for each genome.

			GeneMarkS		Glimmer		Prodigal		GeneMarkS-2	
Species	RefSeq ID	start class	FN	FP	FN	FP	FN	FP	FN	FP
A. fermentans	NC_013740	1	6	275	18	168	5	39	1	124
A. pernix	NC_000854	1	6	130	9	416	9	61	3	19
A. colombiense	NC_014011	1	3	644	10	1351	3	120	3	66
A. variabilis	NC_007413	1	7	178	7	416	8	153	7	56
A. phagocytophilum	NC_007797	1	9	366	13	1151	8	337	11	141
A. aeolicus	NC_000918	1	3	613	13	641	4	68	1	161
A. FB24	NC_008541	1	10	52	21	277	5	152	4	19
B. anthracis	NC_007530	1	3	126	4	223	5	39	4	21
B. anthracis	NC_005945	1	2	135	2	228	5	44	3	21
B. cereus	NC_003909	1	8	159	10	240	12	43	7	21
B. cereus	NC_004722	1	6	119	13	223	15	39	9	17
B. halodurans	NC_002570	1	6	287	12	612	13	104	6	60
B. subtilis	NC_014976	1	5	440	10	794	8	126	2	122
B. thetaiotaomicron	NC_004663	1	6	323	14	291	9	357	4	28
B. longum	NC_004307	1	8	168	6	202	8	176	3	37
B. faecium	NC_013172	1	2	24	10	221	4	324	1	1
C. maquilingensis	NC_009954	3	3	514	4	360	1	83	1	81
C. Nitrospira	NC_014355	1	18	249	23	766	16	328	11	64
C. Protochlamydia	NC_005861	1	2	306	5	731	12	63	3	53
C. crescentus	NC_002696	1	15	56	19	78	17	101	15	30
C. flavigena	NC_014151	1	6	47	23	634	1	359	1	10
C. limicola	NC_010803	2	7	700	10	1045	10	444	8	154
C. tepidum	NC_002932	2	11	488	14	330	15	158	14	230
C. aurantiacus	NC_010175	1	17	187	26	375	6	385	8	39
C. acetobutylicum	NC_003030	1	2	52	2	261	4	15	3	17
C. thermocellum	NC_009012	1	3	266	9	582	2	56	0	85
C. burnetii	NC_002971	1	4	437	11	1213	8	174	7	82
C. curtum	NC_013170	1	3	282	12	516	5	176	3	57
C. ATCC	NC_010546	1	5	105	3	234	10	59	4	22
D. desulfuricans	NC_013939	1	2	150	0	422	1	24	0	53
D. deserti	NC_012526	1	14	167	12	420	10	199	6	75
D. radiodurans	NC_001263	1	24	43	26	68	13	103	6	9
D. alaskensis	NC_007519	1	10	772	35	1454	5	391	2	315
D. vulgaris	NC_002937	1	21	107	21	257	11	365	11	33
E. faecalis	NC_004668	1	0	188	2	253	1	30	0	37
E. tasmaniensis	NC_010694	1	8	334	10	580	5	222	5	138
E. coli	NC_004431	1	21	258	27	354	27	175	16	91
E. coli	NC_002655	1	26	297	32	464	31	194	14	115

F. nucleatum	NC_003454	1	1	99	2	172	5	2	3	16
G. aurantiaca	NC_012489	1	8	152	12	435	5	391	6	48
G. sulfurreducens	NC_002939	1	22	100	7	271	3	121	7	38
G. uraniireducens	NC_009483	1	17	284	24	546	20	177	8	108
H. ducreyi	NC_002940	1	2	140	5	272	3	38	1	32
H. influenzae	NC_000907	1	2	168	7	188	3	26	5	37
H. somnus	NC_010519	1	2	158	1	220	2	46	2	27
H. salinarum	NC_010364	3	17	23	10	174	8	109	2	4
H. borinquense	NC_014729	3	3	161	6	345	7	195	1	32
H. utahensis	NC_013158	3	20	104	2	494	5	201	3	31
H. hepaticus	NC_004917	1	0	527	0	546	2	42	0	90
H. pylori	NC_000915	1	1	1047	4	614	5	54	2	106
H. pylori	NC_000921	1	4	967	6	633	8	52	5	86
H. butylicus	NC_008818	1	1	228	2	670	1	189	1	71
K. radiotolerans	NC_009664	1	6	45	38	447	4	349	2	7
L. johnsonii	NC_005362	1	3	89	4	161	0	27	2	14
L. lactis	NC_002662	1	7	147	5	211	8	20	4	28
L. pneumophila	NC_002942	1	4	379	8	951	6	134	5	92
L. interrogans	NC_005823	1	2	544	8	846	4	88	2	63
L. monocytogenes	NC_002973	1	2	160	1	250	2	39	1	30
M. florum	NC_006055	1	1	68	3	60	1	11	1	8
M. loti	NC_002678	1	40	229	49	528	44	216	24	159
M. jannaschii	NC_000909	1	3	82	2	150	3	24	4	14
M. burtonii	NC_007955	1	8	178	11	434	10	97	3	35
M. acetivorans	NC_003552	1	18	467	39	885	21	133	26	104
M. barkeri	NC_007355	1	15	267	17	693	16	96	11	75
M. mazei	NC_003901	1	6	454	17	852	20	98	6	99
M. capsulatus	NC_002977	2	12	176	14	500	9	304	6	124
M. avium	NC_002944	1	33	36	36	255	27	164	15	33
M. bovis	NC_002945	1	34	219	45	789	37	255	32	86
M. leprae	NC_002677	1	6	314	22	3080	10	740	7	222
M. smegmatis	NC_008596	1	17	72	32	204	16	143	8	43
M. tuberculosis	NC_002755	1	33	188	38	719	34	259	27	68
M. tuberculosis	NC_000962	1	19	217	38	805	20	298	20	70
M. agalactiae	NC_013948	1	1	131	0	395	1	29	2	16
M. genitalium	NC_000908	2	3	772	4	1465	1	147	3	245
M. mobile	NC_006908	1	0	338	2	291	1	21	1	31
N. multipartita	NC_013235	1	32	55	31	394	9	212	8	44
N. magadii	NC_013922	3	8	83	2	432	3	236	2	17
N. pharaonis	NC_007426	3	4	99	10	260	4	181	3	20
N. meningitidis	NC_010120	1	17	604	19	590	17	196	13	216
N. meningitidis	NC_003112	1	8	680	14	595	8	207	8	238
N. dassonvillei	NC_014210	1	5	16	8	186	2	343	4	3
N. PCC	NC_003272	1	11	179	13	380	11	160	7	51
N. punctiforme	NC_010628	1	10	228	13	514	8	187	7	70
P. multocida	NC_002663	1	1	246	3	243	1	47	2	34
P. marina	NC_012440	1	0	254	0	309	1	8	0	81
P. staleyi	NC_013720	1	8	435	15	831	5	401	8	145
P. marinus	NC_005042	2	4	343	9	892	3	207	4	81
P. marinus	NC_005072	2	2	196	5	446	4	109	5	54
P. aeruginosa	NC_002516	1	22	44	22	78	6	159	1	39
P. putida	NC_002947	1	30	106	41	223	15	131	14	50
P. syringae	NC_004578	1	34	205	44	335	30	200	8	121
P. aerophilum	NC_003364	3	6	784	8	891	3	275	5	82
P. neutrophilum	NC_010525	3	11	217	10	268	0	91	1	33
P. abyssi	NC_000868	1	1	301	11	421	4	60	1	91

P. furiosus	NC_003413	1	8	374	9	746	12	80	6	92
P. horikoshii	NC_000961	1	4	297	7	642	6	54	3	101
P. yayanosii	NC_015680	1	2	265	0	506	3	59	2	86
R. capsulatus	NC_014034	1	10	44	10	144	6	257	4	34
R. sphaeroides	NC_007493	1	12	26	17	120	9	311	3	3
R. conorii	NC_003103	2	14	120	9	462	14	118	17	43
R. typhi	NC_006142	1	2	61	1	216	0	78	1	9
R. castenholzii	NC_009767	1	22	303	30	443	8	606	4	56
S. viridis	NC_013159	1	7	104	9	452	2	309	3	29
S. enterica	NC_006905	1	15	265	22	439	25	203	7	87
S. enterica	NC_010102	1	16	234	12	413	11	202	4	82
S. keddieii	NC_013521	1	8	69	20	835	2	250	3	10
S. flexneri	NC_004337	1	14	313	119	321	96	161	3	104
S. meliloti	NC_015590	1	11	221	23	376	8	217	9	121
S. heliotrinireducens	NC_013165	1	3	168	8	294	1	169	0	71
S. nassauensis	NC_013947	1	30	44	38	133	13	115	6	14
S. aureus	NC_013450	1	2	52	2	124	7	32	4	9
S. epidermidis	NC_004461	1	5	42	2	128	8	27	8	7
S. pneumoniae	NC_003098	1	4	223	6	300	4	40	3	33
S. pneumoniae	NC_003028	1	6	193	11	274	10	44	7	30
S. pyogenes	NC_002737	1	4	202	4	370	5	51	2	31
S. pyogenes	NC_004070	1	2	204	5	351	5	50	3	29
S. coelicolor	NC_003888	1	28	17	43	300	22	254	16	2
S. acidocaldarius	NC_007181	3	11	197	15	372	9	165	9	45
S. solfataricus	NC_002754	3	24	207	26	503	14	163	11	45
S. tokodaii	NC_003106	3	6	169	15	437	1	104	4	42
S. CC9311	NC_008319	2	16	737	19	1740	10	649	5	199
S. PCC	NC_010475	2	6	707	8	555	2	170	2	150
S. PCC	NC_000911	2	24	754	28	869	16	190	16	138
S. fumaroxidans	NC_008554	1	13	549	12	1369	7	354	5	145
T. bispora	NC_014165	1	11	33	16	496	2	260	6	19
T. gammatolerans	NC_012804	1	2	224	7	301	4	19	1	80
T. kodakarensis	NC_006624	1	1	245	9	271	4	28	1	86
T. onnurineus	NC_011529	1	3	242	8	280	4	56	2	62
T. sibiricus	NC_012883	1	2	416	1	915	12	92	1	52
T. indicus	NC_015681	1	6	731	10	1289	3	87	3	98
T. pendens	NC_008698	3	12	250	8	239	5	36	2	62
T. acidophilum	NC_002578	3	8	500	14	451	8	134	7	79
T. uzoniensis	NC_015315	3	6	261	0	327	2	48	2	42
T. maritima	NC_000853	1	2	575	10	894	4	98	7	145
T. denticola	NC_002967	1	4	509	9	817	8	53	4	102
T. pallidum	NC_000919	1	10	1319	7	2136	7	612	16	46
T. radiovictrix	NC_014221	1	19	37	13	185	11	39	5	9
V. cholerae	NC_002505	1	4	277	9	383	4	141	4	75
V. fischeri	NC_006840	1	1	126	2	132	2	48	3	19
V. distributa	NC_014537	3	3	303	3	195	1	185	1	42
X. campestris	NC_003902	1	28	119	23	467	16	268	10	76
X. oryzae	NC_006834	1	74	292	23	655	39	317	23	179
X. cellulosilytica	NC_013530	1	3	39	16	445	1	265	2	6
X. fastidiosa	NC_002488	1	22	324	17	835	19	468	12	110
Y. pestis	NC_004088	1	15	216	17	424	18	152	10	56

Supplementary Table 2 The test set of genomes with peptide data. We downloaded data of peptides from 63 species generated by the Pacific Northwest National Laboratory (PNNL) (Venter, et al., 2011).

The results of peptides/spectrum mapping to genomes were provided as GFF format files with start and end coordinates of peptides. From this set we removed data related to five species (marked in red). We used the peptide coordinates to find the minimal length ORF that spans the mapped region (from the nearest in-frame upstream start codon to the nearest in-frame downstream stop codon).

Species	RefSeq Accession	GC% of genome	No. of peptide	No. of peptide-supported ORFs	No. of peptides with no upstream start	No. of peptide-supported ORFs with no valid start
Acidiphilium cryptum	NC_009484	68	7,693	1,138	0	0
Actinosynnema mirum	NC_013093	73.7	12,705	1,444	2	2
Anabaena variabilis	NC_007413	41.4	27,450	2,381	18	10
Anaplasma phagocytophilum	NC_007797	41.6	2,112	292	8	1
Arthrobacter FB24	NC_008541	65.5	43,365	2,678	23	18
Bacillus anthracis	NC_005945	35.4	23,557	2,026	6	6
Borrelia burgdorferi	NC_001318	28.6	14,435	640	13	6
Brachybacterium faecium	NC_013172	72	23,820	1,689	16	4
Bradyrhizobium japonicum	NC_004463	64.1	8,955	1,533	0	0
Burkholderia mallei	NC_008785	68.1	21,786	1,250	23	6
Candidatus Pelagibacter	NC_007205	29.7	16,796	1,028	6	2
Caulobacter crescentus	NC_011916	67.2	53,906	2,603	21	19
Cellulomonas flavigena	NC_014151	74.3	44,654	2,110	5	4
Chlorobium tepidum	NC_002932	56.5	23,192	1,473	9	7
Chloroflexus aurantiacus	NC_010175	56.7	39,336	2,357	7	7
Clostridium thermocellum	NC_009012	39	2,186	232	7	1
Cryptobacterium curtum	NC_013170	50.9	13,530	990	8	7
Cyanothece ATCC	NC_010546	37.9	30,917	2,100	22	11
Cyanothece PCC	NC_011884	50.8	14,517	1,302	0	0
Cyanothece PCC	NC_014501	40.2	3,774	330	0	0
Cyanothece PCC	NC_013161	39.8	1,925	220	2	2
Deinococcus radiodurans	NC_001263	67	29,639	1,639	63	29
Desulfovibrio alaskensis	NC_007519	57.8	41,089	2,246	15	12
Desulfovibrio vulgaris	NC_002937	63.1	29,459	2,051	18	16
Ehrlichia chaffeensis	NC_007799	30.1	2,786	370	1	1
Escherichia coli	NC_002655	50.4	17,294	1,673	6	4
Geobacter metallireducens	NC_007517	59.5	25,970	2,121	7	7
Geobacter uraniireducens	NC_009483	54.2	35,911	2,259	12	8
Halogeometricum borinquense	NC_014729	61.1	11,134	1,345	1	1
Halorhabdus utahensis	NC_013158	62.9	12,034	1,362	6	4
Heliobacterium modesticaldum	NC_010337	57	20,304	1,348	10	6
Kineococcus radiotolerans	NC_009664	74.4	38,314	2,340	19	14
Leptospira interrogans	NC_005823	35	15,079	1,687	16	6
Methanosarcina barkeri	NC_007355	39.3	21,358	1,513	50	27
Methanospirillum hungatei	NC_007796	45.1	24,444	1,306	10	6
Mycobacterium tuberculosis	NC_000962	65.6	22,933	2,192	6	6
Nakamurella multipartita	NC_013235	70.9	22,073	1,610	2	2
Nocardiopsis dassonvillei	NC_014210	72.8	12,563	1,212	2	1
Novosphingobium aromaticivorans	NC_007794	65.2	9,712	1,033	0	0
Pelobacter carbinolicus	NC_007498	55.1	4,156	572	0	0
Prochlorococcus marinus	NC_005072	30.8	16,438	1,106	14	7
Rhodobacter capsulatus	NC_014034	66.6	50,614	2,413	24	16
Rhodopseudomonas palustris	NC_005296	65	23,376	2,059	5	4
Roseiflexus castenholzii	NC_009767	60.7	28,131	2,075	3	2
Saccharomonospora viridis	NC_013159	67.3	12,860	1,418	3	3

Salmonella enterica	NC_004631	52.1	22,444	1,931	5	5
Salmonella enterica	NC_003197	52.2	21,356	1,955	13	12
Sanguibacter keddieii	NC_013521	71.9	18,292	1,773	6	4
Slackia heliotrinireducens	NC_013165	60.2	17,454	1,260	2	1
Stackebrandtia nassauensis	NC_013947	68.1	19,901	1,631	4	3
Streptococcus pyogenes	NC_002737	38.5	5,879	651	0	0
Synechococcus PCC	NC_010475	49.6	27,140	1,957	20	14
Synechocystis PCC	NC_000911	47.7	20,806	1,685	6	5
Syntrophobacter fumaroxidans	NC_008554	59.9	19,677	1,395	3	3
Thermobispora bispora	NC_014165	72.4	7,216	968	0	0
Thermosynechococcus elongatus	NC_004113	53.9	278	40	0	0
Xylanimonas cellulosilytica	NC_013530	72.5	40,245	2,134	22	11
Yersinia pestis	NC_004088	47.6	30,688	1,271	7	5
Removed genomes:	**RefSeq Accession**	**GC% of genome**	**No. of peptide**	**No. of peptides containing stop codon**		
Escherichia coli	NC_000913	50.8	38485	10829		
Geobacter sulfurreducens	NC_002939	60.9	27613	7196		
Rhodobacter sphaeroides	NC_007493	69	41707	6104		
Magnetospirillum magneticum	NC_007626	65.1	0	0		
Cenarchaeum symbiosum	NC_014820	57.4	2	0		

REFERENCES

Adams, M.D., *et al.* (1991) Complementary-DNA Sequencing - Expressed Sequence Tags and Human Genome Project, *Science*, **252**, 1651-1656.

Aivaliotis, M., *et al.* (2007) Large-scale identification of N-terminal peptides in the halophilic archaea Halobacterium salinarum and Natronomonas pharaonis, *Journal of proteome research*, **6**, 2195-2204.

Altschul, S.F., *et al.* (1990) Basic local alignment search tool, *Journal of molecular biology*, **215**, 403-410.

Altschul, S.F., *et al.* (1997) Gapped BLAST and PSI-BLAST: a new generation of protein database search programs, *Nucleic acids research*, **25**, 3389-3402.

Antonov, I., Baranov, P. and Borodovsky, M. (2013) GeneTack database: genes with frameshifts in prokaryotic genomes and eukaryotic mRNA sequences, *Nucleic acids research*, **41**, D152-D156.

Antonov, I. and Borodovsky, M. (2010) Genetack: frameshift identification in protein-coding sequences by the Viterbi algorithm, *Journal of bioinformatics and computational biology*, **8**, 535-551.

Antonov, I., *et al.* (2013) Identification of the nature of reading frame transitions observed in prokaryotic genomes, *Nucleic acids research*, **41**, 6514-6530.

Badger, J.H. and Olsen, G.J. (1999) CRITICA: Coding region identification tool invoking comparative analysis, *Molecular biology and evolution*, **16**, 512-524.

Baranov, P.V., *et al.* (2005) Transcriptional slippage in bacteria: distribution in sequenced genomes and utilization in IS element gene expression, *Genome biology*, **6**, R25.

Bateman, A., *et al.* (2004) The Pfam protein families database, *Nucleic acids research*, **32**, D138-141.

Besemer, J. and Borodovsky, M. (1999) Heuristic approach to deriving models for gene finding, *Nucleic acids research*, **27**, 3911-3920.

Besemer, J. and Borodovsky, M. (2005) GeneMark: web software for gene finding in prokaryotes, eukaryotes and viruses, *Nucleic acids research*, **33**, W451-454.

Besemer, J., Lomsadze, A. and Borodovsky, M. (2001) GeneMarkS: a self-training method for prediction of gene starts in microbial genomes. Implications for finding sequence motifs in regulatory regions, *Nucleic acids research*, **29**, 2607-2618.

Birney, E., Thompson, J.D. and Gibson, T.J. (1996) PairWise and SearchWise: Finding the optimal alignment in a simultaneous comparison of a protein profile against all DNA translation frames, *Nucleic acids research*, **24**, 2730-2739.

Blattner, F.R., *et al.* (1997) The complete genome sequence of Escherichia coli K-12, *Science*, **277**, 1453-1462.

Boeckmann, B., *et al.* (2003) The SWISS-PROT protein knowledgebase and its supplement TrEMBL in 2003, *Nucleic acids research*, **31**, 365-370.

Boguski, M.S. (1995) The Turning-Point in Genome Research, *Trends in biochemical sciences*, **20**, 295-296.

Booijink, C.C.G.M., *et al.* (2010) Metatranscriptome Analysis of the Human Fecal Microbiota Reveals Subject-Specific Expression Profiles, with Genes Encoding Proteins Involved in Carbohydrate Metabolism Being Dominantly Expressed, *Applied and environmental microbiology*, **76**, 5533-5540.

Borodovsky, M. and Mcininch, J. (1993) Genmark - Parallel Gene Recognition for Both DNA Strands, *Computers & chemistry*, **17**, 123-133.

Borodovsky, M. and McIninch, J. (1993) GENMARK: parallel gene recognition for both DNA strands, *Computers & chemistry*, **17**, 123-133.

Borodovsky, M., *et al.* (1986) Statistical features in the Escherichia coli genome functional primary structure. III. Computer recognition of protein coding regions, *Mol. Biol.*, **20**, 1144-1150.

Bult, C.J., *et al.* (1996) Complete genome sequence of the methanogenic archaeon, Methanococcus jannaschii, *Science*, **273**, 1058-1073.

Chen, S.L., *et al.* (2004) Codon usage between genomes is constrained by genome-wide mutational processes, *Proceedings of the National Academy of Sciences of the United States of America*, **101**, 3480-3485.

Claverie, J.M. (1993) Detecting Frame Shifts by Amino-Acid-Sequence Comparison, *Journal of molecular biology*, **234**, 1140-1157.

Decatur, W.A. and Fournier, M.J. (2003) RNA-guided nucleotide modification of ribosomal and other RNAs, *The Journal of biological chemistry*, **278**, 695-698.

Delcher, A.L., *et al.* (2007) Identifying bacterial genes and endosymbiont DNA with Glimmer, *Bioinformatics*, **23**, 673-679.

Delcher, A.L., *et al.* (1999) Improved microbial gene identification with GLIMMER, *Nucleic acids research*, **27**, 4636-4641.

Dyer, D., *et al.* (2011) Sequential shrink photolithography for plastic microlens arrays, *Applied Physics Letters*, **99**.

Forsberg, K.J., *et al.* (2014) Bacterial phylogeny structures soil resistomes across habitats, *Nature*, **509**, 612-+.

Fraser, C.M., *et al.* (1995) The minimal gene complement of Mycoplasma genitalium, *Science*, **270**, 397-403.

Frias-Lopez, J., *et al.* (2008) Microbial community gene expression in ocean surface waters, *PNAS*, **105**, 3805-3810.

Frishman, D., *et al.* (1998) Combining diverse evidence for gene recognition in completely sequenced bacterial genomes (vol 26, pg 2941, 1998), *Nucleic Acids Res*, **26**, U7-U7.

Fritsch, C., *et al.* (2012) Genome-wide search for novel human uORFs and N-terminal protein extensions using ribosomal footprinting, *Genome research*, **22**, 2208-2218.

Fukunishi, Y. and Hayashizaki, Y. (2001) Amino acid translation program for full-length cDNA sequences with frameshift errors, *Physiological genomics*, **5**, 81-87.

Galperin, M.Y., *et al.* (2015) Expanded microbial genome coverage and improved protein family annotation in the COG database, *Nucleic acids research*, **43**, D261-269.

Garber, M., *et al.* (2011) Computational methods for transcriptome annotation and quantification using RNA-seq, *Nature methods*, **8**, 469-477.

Gerashchenko, M.V. and Gladyshev, V.N. (2015) Translation inhibitors cause abnormalities in ribosome profiling experiments, *Nucleic acids research*, **42**, e134.

Grabherr, M.G., *et al.* (2011) Full-length transcriptome assembly from RNA-Seq data without a reference genome, *Nat Biotechnol*, **29**, 644-652.

Guan, X.J. and Uberbacher, E.C. (1996) Alignments of DNA and protein sequences containing frameshift errors, *Computer Applications in the Biosciences*, **12**, 31-40.

Guttman, M., *et al.* (2010) Ab initio reconstruction of cell type-specific transcriptomes in mouse reveals the conserved multi-exonic structure of lincRNAs, *Nat Biotechnol*, **28**, 503-510.

Haas, B.J., *et al.* (2013) De novo transcript sequence reconstruction from RNA-seq using the Trinity platform for reference generation and analysis, *Nat Protoc*, **8**, 1494-1512.

Hayes, W. and Borodovsky, M. (1998) How to Interpret an Anonymous Bacterial Genome: Machine Learning Approach to Gene Identification, *Genome Res.*, **8**, 1154-1171.

Heidorn, T., *et al.* (2011) Synthetic Biology in Cyanobacteria: Engineering and Analyzing Novel Functions, *Methods in Enzymology, Vol 497: Synthetic Biology, Methods for Part/Device Characterization and Chassis Engineering, Pt A*, **497**, 539-579.

Himmelreich, R., *et al.* (1996) Complete sequence analysis of the genome of the bacterium Mycoplasma pneumoniae, *Nucleic acids research*, **24**, 4420-4449.

Hoff, K.J. (2009) The effect of sequencing errors on metagenomic gene prediction, *BMC genomics*, **10**, 520.

Hoff, K.J., *et al.* (2015) BRAKER1: Unsupervised RNA-Seq-Based Genome Annotation with GeneMark-ET and AUGUSTUS, *Bioinformatics*.

Huang, X. (1996) Fast comparison of a DNA sequence with a protein sequence database, *Microb Comp Genomics*, **1**, 281-291.

Hyatt, D., *et al.* (2010) Prodigal: prokaryotic gene recognition and translation initiation site identification, *BMC bioinformatics*, **11**, 119.

Ingolia, N.T., *et al.* (2009) Genome-Wide Analysis in Vivo of Translation with Nucleotide Resolution Using Ribosome Profiling, *Science*, **324**, 218-223.

Ingolia, N.T., Lareau, L.F. and Weissman, J.S. (2011) Ribosome Profiling of Mouse Embryonic Stem Cells Reveals the Complexity and Dynamics of Mammalian Proteomes, *Cell*, **147**, 789-802.

Iseli, C., Jongeneel, C.V. and Bucher, P. (1999) ESTScan: a program for detecting, evaluating, and reconstructing potential coding regions in EST sequences, *Proceedings / ... International Conference on Intelligent Systems for Molecular Biology ; ISMB. International Conference on Intelligent Systems for Molecular Biology*, **99**, 138-148.

Ismail, W.M., Ye, Y. and Tang, H. (2014) Gene finding in metatranscriptomic sequences, *BMC bioinformatics*, **15 Suppl 9**, S8.

Karlsson, F.H., *et al.* (2013) Gut metagenome in European women with normal, impaired and diabetic glucose control, *Nature*, **498**, 99-+.

Kelley, D.R., *et al.* (2012) Gene prediction with Glimmer for metagenomic sequences augmented by classification and clustering, *Nucleic acids research*, **40**, e9.

Kislyuk, A., *et al.* (2009) Frameshift detection in prokaryotic genomic sequences, *International journal of bioinformatics research and applications*, **5**, 458-477.

Klenk, H.P., *et al.* (1997) The complete genome sequence of the hyperthermophilic, sulphate-reducing archaeon Archaeoglobus fulgidus, *Nature*, **390**, 364-370.

Kong, L., *et al.* (2007) CPC: assess the protein-coding potential of transcripts using sequence features and support vector machine, *Nucleic acids research*, **35**, W345-349.

Kozak, M. (1987) An Analysis of 5'-Noncoding Sequences from 699 Vertebrate Messenger-Rnas, *Nucleic acids research*, **15**, 8125-8148.

Krause, L., *et al.* (2007) GISMO--gene identification using a support vector machine for ORF classification, *Nucleic acids research*, **35**, 540-549.

Kunin, V., *et al.* (2008) A Bioinformatician's Guide to Metagenomics, *Microbiol Mol Biol R*, **72**, 557-578.

Kunst, F., *et al.* (1997) The complete genome sequence of the gram-positive bacterium Bacillus subtilis, *Nature*, **390**, 249-256.

Larsen, T. and Krogh, A. (2003) EasyGene - a prokaryotic gene finder that ranks ORFs by statistical significance, *BMC bioinformatics*, **4**.

Lawrence, C.E., *et al.* (1993) Detecting subtle sequence signals: a Gibbs sampling strategy for multiple alignment, *Science*, **262**, 208-214.

Lee, S., *et al.* (2012) Global mapping of translation initiation sites in mammalian cells at single-nucleotide resolution, *Proceedings of the National Academy of Sciences of the United States of America*, **109**, E2424-E2432.

Levin, J.Z., *et al.* (2010) Comprehensive comparative analysis of strand-specific RNA sequencing methods, *Nat Methods*, **7**, 709-U767.

Lew, J.M., *et al.* (2011) TubercuList-10 years after, *Tuberculosis*, **91**, 1-7.

Li, J.J., *et al.* (2011) Sparse linear modeling of next-generation mRNA sequencing (RNA-Seq) data for isoform discovery and abundance estimation, *Proceedings of the National Academy of Sciences of the United States of America*, **108**, 19867-19872.

Liu, J.F., Gough, J. and Rost, B. (2006) Distinguishing protein-coding from non-coding RNAs through support vector machines, *Plos Genetics*, **2**, 529-536.

Lomsadze, A., Burns, P.D. and Borodovsky, M. (2014) Integration of mapped RNA-Seq reads into automatic training of eukaryotic gene finding algorithm, *Nucleic acids research*, **42**, e119.

Lukashin, A.V. and Borodovsky, M. (1998) GeneMark.hmm: new solutions for gene finding, *Nucleic acids research*, **26**, 1107-1115.

Lukashin, A.V., Engelbrecht, J. and Brunak, S. (1992) Multiple alignment using simulated annealing: branch point definition in human mRNA splicing, *Nucleic acids research*, **20**, 2511-2516.

Luo, C., *et al.* (2012) Direct comparisons of Illumina vs. Roche 454 sequencing technologies on the same microbial community DNA sample, *PloS one*, **7**, e30087.

Luo, C., *et al.* (2012) Individual genome assembly from complex community short-read metagenomic datasets, *The ISME journal*, **6**, 898-901.

Maas, S. (2012) Posttranscriptional recoding by RNA editing, *Adv Protein Chem Struct Biol*, **86**, 193-224.

Martin, J., *et al.* (2010) Bacillus anthracis genome organization in light of whole transcriptome sequencing, *BMC bioinformatics*, **11**.

Medigue, C., *et al.* (1999) Detecting and analyzing DNA sequencing errors: Toward a higher quality of the Bacillus subtilis genome sequence, *Genome research*, **9**, 1116-1127.

Mezlini, A.M., *et al.* (2013) iReckon: simultaneous isoform discovery and abundance estimation from RNA-seq data, *Genome research*, **23**, 519-529.

Min, X.J., *et al.* (2005) OrfPredictor: predicting protein-coding regions in EST-derived sequences, *Nucleic acids research*, **33**, W677-680.

Mortazavi, A., *et al.* (2008) Mapping and quantifying mammalian transcriptomes by RNA-Seq, *Nature methods*, **5**, 621-628.

Nagaraj, S.H., Gasser, R.B. and Ranganathan, S. (2007) A hitchhiker's guide to expressed sequence tag (EST) analysis, *Briefings in bioinformatics*, **8**, 6-21.

Neuwald, A.F., Liu, J.S. and Lawrence, C.E. (1995) Gibbs motif sampling: detection of bacterial outer membrane protein repeats, *Protein Sci*, **4**, 1618-1632.

Nielsen, H.B., *et al.* (2014) Identification and assembly of genomes and genetic elements in complex metagenomic samples without using reference genomes, *Nature Biotechnology*, **32**, 822-828.

Nielsen, P. and Krogh, A. (2005) Large-scale prokaryotic gene prediction and comparison to genome annotation, *Bioinformatics*, **21**, 4322-4329.

Passalacqua, K.D., *et al.* (2009) Structure and Complexity of a Bacterial Transcriptome, *Journal of bacteriology*, **191**, 3203-3211.

Posfai, J. and Roberts, R.J. (1992) Finding Errors in DNA-Sequences, *Proceedings of the National Academy of Sciences of the United States of America*, **89**, 4698-4702.

Rabiner, L.R. (1989) A Tutorial on Hidden Markov-Models and Selected Applications in Speech Recognition, *Proceedings of the Ieee*, **77**, 257-286.

Reid, I., *et al.* (2014) SnowyOwl: accurate prediction of fungal genes by using RNA-Seq and homology information to select among ab initio models, *BMC bioinformatics*, **15**, 229.

Rho, M., Tang, H. and Ye, Y. (2010) FragGeneScan: predicting genes in short and error-prone reads, *Nucleic acids research*, **38**, e191.

Roberts, A., *et al.* (2011) Identification of novel transcripts in annotated genomes using RNA-Seq, *Bioinformatics*, **27**, 2325-2329.

Roberts, A., *et al.* (2011) Identification of novel transcripts in annotated genomes using RNA-Seq, *Bioinformatics*, **27**, 2325-2329.

Robertson, G., *et al.* (2010) De novo assembly and analysis of RNA-seq data, *Nature methods*, **7**, 909-U962.

Rudd, K.E. (2000) EcoGene: a genome sequence database for Escherichia coli K-12, *Nucleic acids research*, **28**, 60-64.

Salzberg, S.L., *et al.* (1998) Microbial gene identification using interpolated Markov models, *Nucleic acids research*, **26**, 544-548.

Schiex, T., *et al.* (2003) FrameD: A flexible program for quality check and gene prediction in prokaryotic genomes and noisy matured eukaryotic sequences, *Nucleic acids research*, **31**, 3738-3741.

Schulz, M.H., *et al.* (2012) Oases: robust de novo RNA-seq assembly across the dynamic range of expression levels, *Bioinformatics*, **28**, 1086-1092.

Sekhar, M.C., *et al.* (2011) Crossover in domain wall potential polarity as a function of anti-notch geometry, *Journal of Physics D-Applied Physics*, **44**.

Sharma, C.M., *et al.* (2010) The primary transcriptome of the major human pathogen Helicobacter pylori, *Nature*, **464**, 250-255.

Sharma, V., *et al.* (2011) A pilot study of bacterial genes with disrupted ORFs reveals a surprising profusion of protein sequence recoding mediated by ribosomal frameshifting and transcriptional realignment, *Molecular biology and evolution*, **28**, 3195-3211.

Shine, J. and Dalgarno, L. (1974) Identical 3'-terminal octanucleotide sequence in 18S ribosomal ribonucleic acid from different eukaryotes. A proposed role for this sequence in the recognition of terminator codons, *The Biochemical journal*, **141**, 609-615.

Skovgaard, M., *et al.* (2001) On the total number of genes and their length distribution in complete microbial genomes, *Trends in Genetics*, **17**, 425-428.

Slater, G.S. and Birney, E. (2005) Automated generation of heuristics for biological sequence comparison, *BMC bioinformatics*, **6**.

Slupska, M.M., *et al.* (2001) Leaderless transcripts of the crenarchaeal hyperthermophile Pyrobaculum aerophilum, *Journal of molecular biology*, **309**, 347-360.

Smith, H.O., *et al.* (1995) Frequency and distribution of DNA uptake signal sequences in the Haemophilus influenzae Rd genome, *Science*, **269**, 538-540.

Stanke, M., *et al.* (2006) AUGUSTUS: ab initio prediction of alternative transcripts, *Nucleic acids research*, **34**, W435-W439.

Steijger, T., *et al.* (2013) Assessment of transcript reconstruction methods for RNA-seq, *Nature methods*, **10**, 1177-1184.

Tang, S., Antonov, I. and Borodovsky, M. (2013) MetaGeneTack: ab initio detection of frameshifts in metagenomic sequences, *Bioinformatics*, **29**, 114-116.

Tang, S.Y.Y., Antonov, I. and Borodovsky, M. (2013) MetaGeneTack: ab initio detection of frameshifts in metagenomic sequences, *Bioinformatics*, **29**, 114-116.

Tatusov, R.L., *et al.* (2003) The COG database: an updated version includes eukaryotes, *BMC bioinformatics*, **4**, 41.

Tatusov, R.L., Koonin, E.V. and Lipman, D.J. (1997) A genomic perspective on protein families, *Science*, **278**, 631-637.

Torarinsson, E., Klenk, H.P. and Garrett, R.A. (2005) Divergent transcriptional and translational signals in Archaea, *Environmental microbiology*, **7**, 47-54.

Trapnell, C., *et al.* (2013) Differential analysis of gene regulation at transcript resolution with RNA-seq, *Nature Biotechnology*, **31**, 46-+.

Trapnell, C., Pachter, L. and Salzberg, S.L. (2009) TopHat: discovering splice junctions with RNA-Seq, *Bioinformatics*, **25**, 1105-1111.

Trapnell, C., *et al.* (2012) Differential gene and transcript expression analysis of RNA-seq experiments with TopHat and Cufflinks, *Nat Protoc*, **7**, 562-578.

Trapnell, C., *et al.* (2010) Transcript assembly and quantification by RNA-Seq reveals unannotated transcripts and isoform switching during cell differentiation, *Nat Biotechnol*, **28**, 511-515.

Turnbaugh, P.J., *et al.* (2010) Organismal, genetic, and transcriptional variation in the deeply sequenced gut microbiomes of identical twins, *PNAS*, **107**, 7503-7508.

Turner, T.R., *et al.* (2013) Comparative metatranscriptomics reveals kingdom level changes in the rhizosphere microbiome of plants, *Isme Journal*, **7**, 2248-2258.

Tyakht, A.V., *et al.* (2013) Human gut microbiota community structures in urban and rural populations in Russia, *Nature communications*, **4**.

Venter, E., Smith, R.D. and Payne, S.H. (2011) Proteogenomic analysis of bacteria and archaea: a 46 organism case study, *PloS one*, **6**, e27587.

Vivancos, A.P., *et al.* (2010) Strand-specific deep sequencing of the transcriptome, *Genome research*, **20**, 989-999.

Wang, K., *et al.* (2010) MapSplice: accurate mapping of RNA-seq reads for splice junction discovery, *Nucleic acids research*, **38**, e178.

Wang, Z., Gerstein, M. and Snyder, M. (2009) RNA-Seq: a revolutionary tool for transcriptomics, *Nature reviews. Genetics*, **10**, 57-63.

Wernegreen, J.J., Kauppinen, S.N. and Degnan, P.H. (2010) Slip into Something More Functional: Selection Maintains Ancient Frameshifts in Homopolymeric Sequences, *Molecular biology and evolution*, **27**, 833-839.

Wooley, J.C., Godzik, A. and Friedberg, I. (2010) A primer on metagenomics, *PLoS computational biology*, **6**, e1000667.

Wurtzel, O., *et al.* (2010) A single-base resolution map of an archaeal transcriptome, *Genome Res*, **20**, 133-141.

Yamazaki, S., *et al.* (2006) Proteome analysis of an aerobic hyperthermophilic crenarchaeon, Aeropyrum pernix K1, *Mol Cell Proteomics*, **5**, 811-823.

Yoon, S.H., Park, Y.K. and Kim, J.F. (2015) PAIDB v2.0: exploration and analysis of pathogenicity and resistance islands, *Nucleic acids research*, **43**, D624-630.

Zerbino, D.R. and Birney, E. (2008) Velvet: Algorithms for de novo short read assembly using de Bruijn graphs, *Genome research*, **18**, 821-829.

Zhou, J.D. and Rudd, K.E. (2013) EcoGene 3.0, *Nucleic acids research*, **41**, D613-D624.

Zhu, W., Lomsadze, A. and Borodovsky, M. (2010) Ab initio gene identification in metagenomic sequences, *Nucleic acids research*, **38**, e132.