Gene prediction in metagenomic sequencing reads

PhD Thesis

in partial fulfilment of the requirements
for the degree "Doctor of Philosophy (PhD)"
in the Molecular Biology Program
at the Georg August University Göttingen,
Faculty of Biology

submitted by

Katharina Jasmin Hoff

born in

Siegburg

Göttingen 2009

Thesis Supervisor:
Dr. Peter Meinicke

Doctoral Committee:
Prof. Dr. Burkhard Morgenstern (1^{st} Referee)
Prof. Dr. Wolfgang Liebl (2^{nd} Referee)
Prof. Dr. Lutz Walter

Affidavit

I hereby insure that this PhD thesis has been written independently and with no other sources and aids than quoted.

Katharina J. Hoff

August, 2009

Göttingen, Germany

List of Publications

Papers in Peer Reviewed Journals

- K. J. Hoff, M. Tech, T. Lingner, R. Daniel, B. Morgenstern, P. Meinicke
  Gene prediction in metagenomic fragments: a large scale machine learning approach
  BMC Bioinformatics 2008 9:217

- K. J. Hoff, T. Lingner, P. Meinicke, M. Tech
  Orphelia: predicting genes in metagenomic sequencing reads
  Nucleic Acids Research 2009 37:W101-W105

- K. J. Hoff.
  The effect of sequencing errors on metagenomic gene prediction.
  BMC Genomics 2009 10:520

Posters at International Conferences

- K. J. Hoff, M. Tech, P. Meinicke
  Gene prediction in metagenomic DNA fragments with machine learning techniques
  Genomes 2008, Paris, France

- K. J. Hoff, M. Tech, P. Meinicke
  Predicting genes on metagenomic pyrosequencing reads with machine learning techniques
  German Conference on Bioinformatics 2008, Dresden, Germany

- K. J. Hoff, M. Tech, T. Lingner, R. Daniel, B. Morgenstern, P. Meinicke
  Gene prediction in metagenomic DNA fragments
  Horizons in Molecular Biology 2008, Göttingen, Germany

- K. J. Hoff, M. Tech, P. Meinicke
  Predicting genes in metagenomic DNA fragments with high specificity using machine learning techniques
  European Conference on Computational Biology 2008, Cagliari, Italy

- K. J. Hoff, M. Tech, P. Meinicke
  Predicting genes in short metagenomic sequencing reads with high specificity
  Metagenomics 2008, San Diego, United States of America

- K. J. Hoff, F. Schreiber, M. Tech, P. Meinicke
  The effect of sequencing errors on metagenomic gene prediction
  International Conference on Intelligent Systems for Molecular Biology/European Conference on Computational Biology 2009, Stockholm, Sweden

Contents

1 General Introduction **3**

1.1 Assembly . . . 5

1.2 Sequence Similarity . . . 6

1.3 Statistical Models . . . 6

1.3.1 GeneMark with Heuristic Models . . . 7

1.3.2 MetaGene and MetaGeneAnnotator . . . 7

1.4 Sequencing Techniques . . . 8

1.4.1 Sanger Sequencing . . . 8

1.4.2 Pyrosequencing . . . 9

1.5 Objective . . . 10

2 List of Publications **12**

3 Gene Prediction in Metagenomic Fragments: a Large Scale Machine Learning Approach **13**

4 Orphelia: Predicting Genes in Metagenomic Sequencing Reads **28**

5 The Effect of Sequencing Errors on Metagenomic Gene Prediction **34**

6 General Discussion **56**

6.1 Training Data . . . 57

6.2 More Features of Protein Coding Genes . . . 57

6.3 Applicability to Real Data . . . 58

7 Summary and Conclusions **60**

Acknowledgements

I much appreciate the support of my supervisor Peter Meinicke throughout the last three years. He has guided me to this thesis topic and coordinated the close collaboration with Maike Tech, to whom I am also very grateful for her constant support and many fruitful discussions. I am deeply grateful to Professor Burkhard Morgenstern, who gave me the freedom to work in his department, and who is the first referee of my thesis.

Thanks to Professor Wolfgang Liebl for being the second referee, and thanks to Professor Lutz Walter for support and fruitful discussions. I would also like to thank Rolf Daniel for very helpful discussions about metagenomics.

I thank Thomas Lingner, Rasmus Steinkamp, Fabian Schreiber, Mario Stanke, Markus Fritz, and Falk Hildebrandt for their contributions to the progress of my project - either by sharing their own scripts or by helping me with tedious computer problems.

The coordination office of the Molecular Biology program was extremely helpful during all phases of my PhD studies and I want to thank them for their commitment.

A special thanks goes to my parents Alexandra and Klaus Hoff for motivating me throughout the four years of the MSc/PhD Molecular Biology program. I thank the "Ex-Geists" (Marija Sumakovic, Konstantina Marinoglou, Adema Ribic, Ieva Gailite), and my friends Gabriel Mora-Oberlaender, Achim Werner, Andrew Woehler, and Christian Roschke for great company, friendship and really good parties during the last four years. I also thank Margrit Walter and Horst Kramer for shifting the "work-life-balance" towards "life" every now and then.

Abstract

Gene prediction is an essential step in the annotation of metagenomic sequencing reads. Since most metagenomic reads cannot be assembled into long contigs, specialized gene prediction tools are required for the analysis of short and anonymous DNA fragments.

This work describes the metagenomic gene prediction method 'Orphelia'. It consists of a two-stage machine learning approach. In the first stage, linear discriminants for monocodon usage, dicodon usage and translation initiation sites are used to extract features from DNA sequences. In the second stage, an artificial neural network combines these features with open reading frame length and fragment GC-content to compute the probability that this open reading frame encodes a protein. This probability is used for the classification and scoring of gene candidates. Orphelia is available to the scientific community as an intuitive web server application, and as a command line tool.

Furthermore, a detailed evaluation of gene prediction accuracy of Orphelia and other tools with respect to sequencing errors an read length is presented. It is demonstrated that ESTScan, a tool for sequencing error compensation in eukaryotic expressed sequence tags, outperforms some metagenomic gene prediction tools on reads with high error rates although it was not designed for the task at hand. The integration of error-compensating methods into metagenomic gene prediction tools would be beneficial to improve metagenome annotation quality.

Chapter 1

General Introduction

Prokaryotes are single-cell organisms that lack a cell nucleus and other membrane-bound organelles. They form the biggest group of living organisms on earth and encompass the kingdoms *Bacteria* and *Archaea*. The total number of prokaryotic species is unknown but preliminary estimates allow a glimpse at the dimension of their diversity: Curtis *et al.* (2002) expect 2×10^6 bacterial taxa in the sea, and about 4×10^6 different taxa in soil [1]. Single bacterial or archaeal species are usually scientifically investigated by techniques that require the cultivation of a species under laboratory conditions. The problem here is that only few species are culturable. Even the application of a wide range of available growth conditions has so far led to the successful cultivation of only $\sim$1% of all species [2, 3, 4]. A large fraction of prokaryotic species can therefore not be investigated by conventional methods.

To overcome this problem, a set of molecular techniques has been developed that allows the investigation of microbial genomes without cultivation. Using this so-called metagenomic approach, the genomes of a microbial community are analyzed simultaneously. Genomic material (deoxyribonucleic acid, abbreviated as DNA) is directly isolated from the environment and sequenced. Typically, one out of two sequencing techniques are applied to metagenomes:

- chain termination sequencing, also known as Sanger sequencing [5], or
- pyrosequencing, also named 454 sequencing [6].

The characteristic workflows for each technique are depicted in figure 1 on page 4. In the case of Sanger sequencing, it consists of extracting genomic DNA from a habitate, cloning the obtained DNA fragments into a vector, transforming the vector into a host strain (e.g. *Escherichia coli*), cultivating many transformed hosts with different vector inserts, and finally sequencing the inserts with chain termination technique. Most of the reads obtained

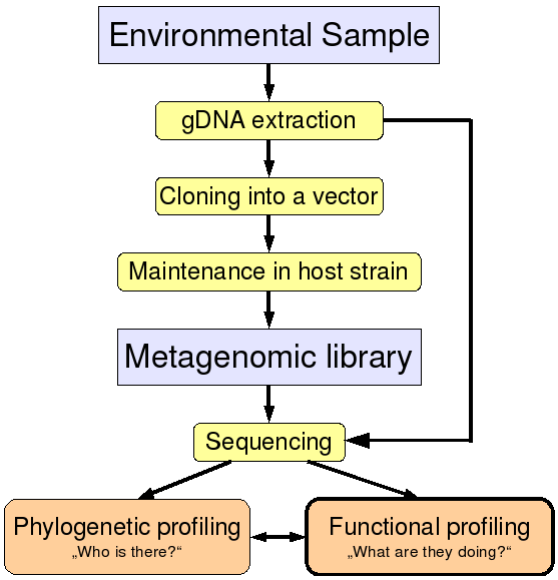


Figure 1.1: Metagenomic workflow for Sanger sequencing (left) and pyrosequencing (right arrow).

this way are ∼700 to 1000 bases (b) long. It is also possible to sequence an insert from both vector ends and construct a longer consensus sequence. However, not all possible inserts can be sequenced. An insert that carries a gene with a promotor e.g. encoding a compound that is toxic for the host cell, will not be found because the host dies after transformation. The resulting effect is named the *cloning bias*. Nevertheless, chain termination sequencing is due to its read length often used for sequencing metagenomes. It has e.g. been applied to a soil metagenome [7], the Sargasso sea metagenome [8], and the hypersaline microbial mat metagenome [9].

The *cloning bias* can be circumvented by pyrosequencing where the extracted environmental genomic DNA is directly sequenced. In comparison to Sanger sequencing, pyrosequencing can be massively parallelized and thus allows the sequencing of much bigger amounts of genomic material. In the beginning, the read length of 454 sequencing was ∼120 b. It has recently increased to ∼400 b. Sanger- and pyrosequencing are described in detail in section 1.4 on pages 8ff.

Regardless of the technique that is used for sequencing a metagenome, the result is a large collection of sequencing reads from several species. The taxonomic origin of each read is unknown, and it is unclear whether and which stretches on the reads carry protein coding genes (PCGs). However, both types of information are crucial for further metagenome analysis, which usually aims at the estimation of a functional and phylogenetic profile of the microbial community under investigation.

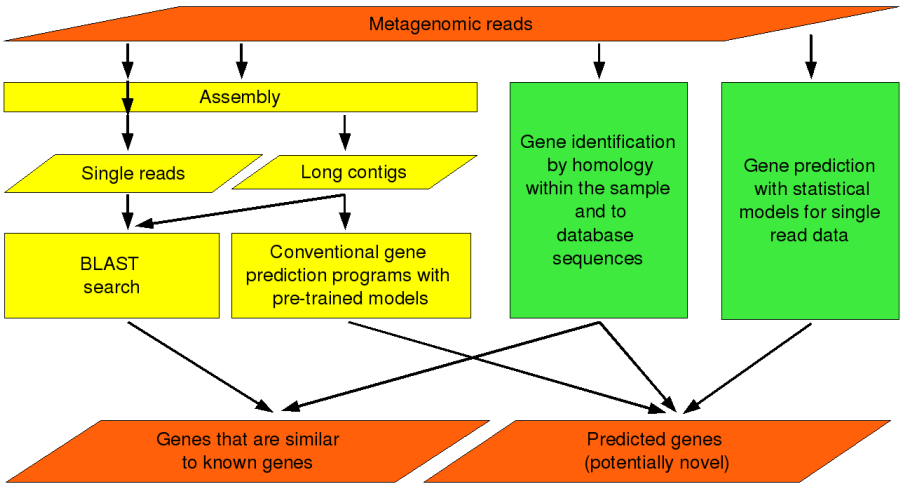


Figure 1.2: Gene prediction workflows in metagenomics. The *conventional* workflow is shown in yellow. Two types of tools that are specialized on metagenomic single read data exist: homology based methods and statistical modelling methods (both shown in green).

This work addresses the problem of identifying PCGs in metagenomic sequencing reads. Figure 1.2 gives an overview on possible approaches, which are described in the following sections in detail.

1.1 Assembly

Assembly is the recruitment of single reads into long stretches of DNA ("contigs") by overlap of reads. In genomics, assembly is usually the first step after genome sequencing, and genes are subsequently identified in long sequences. Assembly is often also applied to metagenomics but an assessment has demonstrated that this is not trivial [10]. One major problem is the reliability of a contig. Due to the number of species in a community it can easily happen that reads from different species that carry parts of homologous genes are assembled into a chimeric contig. With a high sequencing coverage for a single species, this risk decreases and the resulting "long contigs" (>8 knt) are likely to be homogeneous. Thus, long contigs are useful for further analysis with standard genomics tools. However, sequencing with sufficient coverage is only possible for species that dominate the community. In some microbial communities, there are no dominating species (e.g. for the hypersaline microbial mat metagenome, assembly was impossible [9]). In the best case, one ends up with a proportion of long contigs and few single reads. But in most cases, the proportion of single

reads will be large. Therefore, specialized methods for read analysis are needed in order to assess the genetic potential of a community.

1.2 Gene Identification through Sequence Similarity

The most common way of identifying genes in anonymous metagenomic reads is to perform a BLAST search with metagenomic sequences against databases of known proteins [11]. This strategy will reveal already known genes that are present in the metagenome. The problem is that many genes are most probably not present in the databases, yet, due to the bias towards culturable organisms in the databases. Those novel genes cannot be detected with a BLAST search.

Another set of methods relies on the assumption that PCGs are better conserved than noncoding regions in prokaryotic genomes because mutations in PCGs may end fatally for the organism. Therefore, it is possible to find genes by searching for highly conserved sequence regions within a metagenome. Several methods have been developed for this purpose, among them an algorithm that is similar to BLAST [12] and a clustering algorithm on the basis of cd-hit, a cluster program for producing a set of non-redundant representative sequences [13, 14].

All these “intra“ sample homology methods are computationally expensive and thus time consuming. They are not an option for research groups that are not equipped with suitable hardware. However, in comparison to other gene finding methods, most sequence similarity based approaches have the advantage to be relatively robust to sequencing errors.

1.3 Model-based Gene Prediction

The prediction of genes with statistical models is usually fast, and bears the potential to detect novel genes. In general, model-based methods require an initial training phase on data from the target genome in order to adapt to species-specific characteristics of protein coding regions. Several gene prediction tools of this kind have been developed and successfully applied to (near to) complete prokaryotic genomes, e.g. GLIMMER [15] or GeneMark [16]. However, their direct application to metagenomes is difficult because a metagenome contains anonymous reads from more than one species. This makes assembly difficult and thus, the analysis of single reads is often necessary (see section 1.1 on page 5). Only in some habitates, one or few species dominate the microbial community and it is possible to assemble the reads of those species into long contigs. A long contig may offer sufficient training data but the

trained model will in most cases only be applicable to the training contig and some other few contigs that are known to originate from the same species.

A different strategy is to train numerous different models on the genomes of groups of known and closely related species. Reads of a metagenomic sample are *binned* into taxonomic groups (e.g. with Phylopythia [17]), and genes in the reads of a bin are predicted with an appropriate pre-trained model. The limiting factor for gene prediction accuracy here is the binning accuracy, which is not very high for single reads. Another problem is that models cannot be pre-trained for yet unknown taxonomic groups.

To overcome all these problems, several model-based gene prediction methods have been developed for the application to anonymous single read data, e.g. GeneMark with heuristic models [18], MetaGene [19], and MetaGeneAnnotator [20]. The principles of these methods are described in the following sections.

1.3.1 GeneMark with Heuristic Models

Besemer and Borodovsky built heuristic models on the basis of 17 bacterial genomes for usage with the already existing gene prediction programs GeneMark [21] and a combination of GeneMark and GeneMark.hmm [16]. The heuristic models use the relationships between positional nucleotide frequencies and global nucleotide frequencies, e.g. the occurrence frequency of thymine at the first position of a codon in a genome with an overall thymine frequency of 20%. Furthermore, they utilize the relationships between amino acid frequencies and the guanin-cytosin-content (GC-content) of a genome, e.g. the occurrence frequency of proline in genomes with a GC-content of 40%. The suitable model for predicting genes in short and anonymous sequences is derived from their individual GC-contents and nucleotide frequencies.

The heuristic models do not include any differentiation between bacteria or archaea, and they are reported to work well for input sequences above a length of 400 b [18].

1.3.2 MetaGene and MetaGeneAnnotator

MetaGene is a stand-alone gene prediction program with statistical models that were estimated from 116 bacterial and 15 archaeal species to distinguish PCGs and non-coding open reading frames (nORFs). An open reading frame (ORF) is a stretch of DNA that begins with a start codon and ends with an in-frame stop codon. In metagenomic sequencing reads, ORFs frequently exceed the fragment ends. Therefore, also incomplete ORFs are considered (see Figure 1 in chapter 3 for illustration). The core of MetaGene is a ORF scoring system

on the basis of logistic regressions between GC-content and monocodon as well as dicodon frequencies (codon frequencies are illustrated in figure 1.3 on page 8). In addition, MetaGene calculates log-odd ratios for the ORF length and for the distance of a start codon to the left-most start codon. These sub-scores are combined and a dynamic program is used to compute an optimal high scoring combination of ORFs in an input sequence, also taking ORF orientation and distances to neighboring ORFs into account.

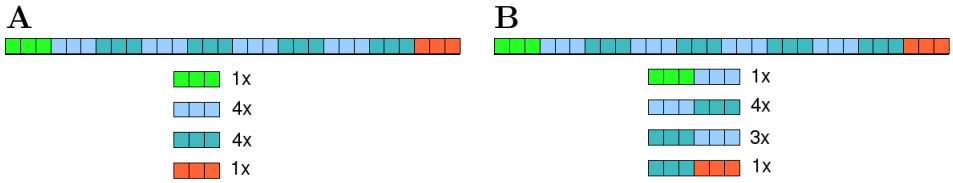


Figure 1.3: Illustration of the features (A) monocodon frequencies and (B) dicodon frequencies. Both features are likely to differ between PCGs and nORFs because the selective pressure on PCGs is higher than on nORFs.

In contrast to the heuristic models of GeneMark, MetaGene has different logistic regression models for Archaea and Bacteria. They are simultaneously applied to a metagenomic sequence and predictions achieved with the highest scoring model are selected. MetaGene is a very sensitive method, i.e. it finds most of the actually existing genes, but it tends to overpredict genes, i.e. often predicts too many genes that do not exist.

The MetaGeneAnnotator is an extension of MetaGene. In addition to the original models, it contains a gene prediction model for prophage genes and a ribosomal binding site model. MetaGeneAnnotator is slightly more accurate than MetaGene on fragmented DNA as it occurs in metagenomes although it was mainly developed for predicting genes in longer sequences, e.g. fosmids.

1.4 Sequencing Techniques

All model based gene prediction programs utilize codon usage as an important feature and are thus susceptible to sequencing errors that alter the codon frequency. For a better understanding of sequencing errors, Sanger and pyrosequencing are described below.

1.4.1 Sanger Sequencing

For Sanger sequencing, a DNA polymerase, a primer, and four types of deoxynucleotide triphosphates (dNTPs) are used to synthesize the complementary strand to the template

sequence. dNTPs usually contain deoxyadenosin triphosphate (dATP), deoxyguanin triphosphate (dGTP), deoxythymin triphosphate (dTTP), and deoxycytosin triphosphate (dCTP). All dNTPs have a 3'hydroxygroup that is required by the DNA polymerase to attach the phosphate group of the following nucleotide during strand synthesis.

Dideoxynucleotide triphosphates (ddNTPs) lack this hydroxygroup. If a ddNTP is incorporated into the sequence, synthesis is terminated. This effect is used in Sanger sequencing, where a small proportion of ddNTPs is added to the sequencing reaction. The result is a number of fragments with different lengths and different terminal ddNTPs (hence also the name "chain termination sequencing"). The ddNTPs are labelled, e.g. radioactively or fluorescently, to allow sequence visualization on a gel. Initially, Sanger and his colleagues split the reaction into four different vials, adding only one type of radioactive ddNTP to each vessel, carrying out four different sequencing reactions. The resulting DNA fragments were visualized in four lanes on a gel and it was possible to reconstruct the DNA sequence by knowing which ddNTP was used for the fragments on each lane. Nowadays, the sequencing reaction is performed in a single vessel with different fluorescently labelled ddNTPs (e.g. ddATP may be green, ddCTP may be red, and so on). The DNA fragments are separated by size through capillary electrophoresis. A laser and a detector are utilized to first excite and then read the fluorphores of each sequence fragment at the end of the capillar, typically producing a chromatogram with four different colors [22].

Sequence quality is low for the first ∼50 b because unreacted primers and unreacted ddNTPs migrate at comparable speed. The following ∼700 to ∼900 b are of higher quality. The most common sequencing errors here are caused by secondary structure formations that increase the migration speed of a fragment through the gel, resulting in deletion errors of the terminal nucleotide at its actual position, and in insertion errors of the same nucleotide at an earlier position in the sequence. For very long sequence fragments, a new problem arises: long fragments need more time to migrate through the gel, thus allowing more time for random diffusion to take place. In addition, the relative mass difference between subsequent fragments decreases, and the number of labelled fragments of a given size decreases, making it increasingly more difficult to differentiate between signal and noise [23]. For this reason, Sanger read length rarely exceeds 1000 b.

1.4.2 Pyrosequencing

Pyrosequencing is a "sequencing by synthesis" technique. A single stranded DNA molecule is hybridized with a primer sequence and incubated with the enzymes DNA polymerase, ATP sulfurylase, luciferase, and apyrase. Additionally, the compounds adenosine-5-phosphosulfate

(APS) and luciferin are added. The building blocks for DNA synthesis, dNTPs, are added to the reaction one type at a time. If the current position in the template sequence is complementary to the dNTP, it will be incorporated, releasing pyrophosphate (PP_i). PP_i and APS are then converted to ATP by ATP sulfurylase. The ATP is further used by luciferase to convert luciferin to oxilucerfin, a reaction that emitts visible light, which is recorded by a camera. Apyrase is added to degrade all unused dNTPs and ATP after a single dNTP flow. In order to avoid constant light production by luciferase, a special form of dNTPs that are not a template for luciferase are used (dNTPαS).

Sequencing errors primarily arise in the case of homopolymer incorporation. The light signal of luciferase is proportional to the amount of dNTP that is incorporated into the synthesized strand but the proportionality ratio is only correct for short homopolymers. Thus, the major type of errors are deletions and insertions in homopolymer stretches. [24]

In contrast to Sanger sequencing, where numerous DNA fragments are synthesized to sequence one piece of DNA, pyrosequencing quality is also affected by DNA polymerase accuracy because in contrast to Sanger sequencing, where many copies of the same template are used for sequence visualization, only few strands are synthesized during 454 sequencing [25].

1.5 Objective

The objective of this work is to support the development of a new and more accurate metagenomic gene prediction method that is based on machine learning techniques. Machine learning generally encompasses methods for the reconstruction of statistical relations or regularities with the help of training examples. Once learned, the statistical relations can be applied for predictions in new data [26]. In the case of gene prediction, biological expertise is required to pre-select features that could potentially be used to discriminate between coding and non-coding regions in metagenomic DNA fragments. Also an accuracy evaluation during all developmental stages is important to select only those features and combinations of features that actually improve prediction quality. For the process of accuracy evaluation, suitable criteria, training data and test data are designed to enable the assessment of gene prediction accuracy.

An important question concerning the applicability of metagenomic gene prediction tools to real data is, to which extent the accuracy of metagenomic gene prediction methods is affected by naturally occurring sequencing errors caused by using different sequencing techniques. This is a largely uninvestigated field for all model based metagenomic gene prediction tools.

Therefore, this thesis also focus on investigating the sequencing error problem.

Chapter 2

List of Publications

The thesis is based on the following original papers:

Chapter 3 — K. J. Hoff, M. Tech, T. Lingner, R. Daniel, B. Morgenstern, P. Meinicke
Gene prediction in metagenomic fragments: a large scale machine learning approach
BMC Bioinformatics 2008 9:217
doi:10.1186/1471-2105-9-217

Chapter 4 — K. J. Hoff, T. Lingner, P. Meinicke, M. Tech
Orphelia: predicting genes in metagenomic sequencing reads
Nucleic Acids Research 2009 37:W101-W105
doi:10.1093/nar/gkp327

Chapter 5 — K. J. Hoff
The effect of sequencing errors on metagenomic gene prediction
BMC Genomics 2009 10:520
doi:10.1186/1471-2164-10-520

Chapter 3

Gene Prediction in Metagenomic Fragments: a Large Scale Machine Learning Approach

Citation

K. J. Hoff, M. Tech, T. Lingner, R. Daniel, B. Morgenstern, P. Meinicke
Gene prediction in metagenomic fragments: a large scale machine learning approach
BMC Bioinformatics 2008 9:217
doi:10.1186/1471-2105-9-217

Original Contribution

Biological expertise with respect to the selection of features for metagenomic gene prediction; implementation of the interface between ORF extraction and machine learning modules; assembly of training and test data sets; evaluation of new method and MetaGene (resulting in figure 2, and tables 2, 3, and 4); manuscript writing (large parts of the introduction, results, discussion, introductory part of methods, and the parts in methods that concern the evaluation procedure and data sets).

BMC Bioinformatics

BioMed Central

Methodology article

Open Access

Gene prediction in metagenomic fragments: A large scale machine learning approach

Katharina J Hoff*[1], Maike Tech[1], Thomas Lingner[1], Rolf Daniel[2], Burkhard Morgenstern[1] and Peter Meinicke[1]

Address: [1]Abteilung Bioinformatik, Georg-August-Universität Göttingen, Goldschmidtstr. 1, 37077 Göttingen, Germany and [2]Abteilung Genomische und Angewandte Mikrobiologie, Georg-August-Universität Göttingen, Goldschmidtstr. 1, 37077 Göttingen, Germany

Email: Katharina J Hoff* - katharina@gobics.de; Maike Tech - maike@gobics.de; Thomas Lingner - thomas@gobics.de; Rolf Daniel - rdaniel@gwdg.de; Burkhard Morgenstern - burkhard@gobics.de; Peter Meinicke - pmeinic@gwdg.de

* Corresponding author

Published: 28 April 2008

BMC Bioinformatics 2008, **9**:217 doi:10.1186/1471-2105-9-217

Received: 9 October 2007
Accepted: 28 April 2008

This article is available from: http://www.biomedcentral.com/1471-2105/9/217

Abstract

Background: Metagenomics is an approach to the characterization of microbial genomes via the direct isolation of genomic sequences from the environment without prior cultivation. The amount of metagenomic sequence data is growing fast while computational methods for metagenome analysis are still in their infancy. In contrast to genomic sequences of single species, which can usually be assembled and analyzed by many available methods, a large proportion of metagenome data remains as unassembled anonymous sequencing reads. One of the aims of all metagenomic sequencing projects is the identification of novel genes. Short length, for example, Sanger sequencing yields on average 700 bp fragments, and unknown phylogenetic origin of most fragments require approaches to gene prediction that are different from the currently available methods for genomes of single species. In particular, the large size of metagenomic samples requires fast and accurate methods with small numbers of false positive predictions.

Results: We introduce a novel gene prediction algorithm for metagenomic fragments based on a two-stage machine learning approach. In the first stage, we use linear discriminants for monocodon usage, dicodon usage and translation initiation sites to extract features from DNA sequences. In the second stage, an artificial neural network combines these features with open reading frame length and fragment GC-content to compute the probability that this open reading frame encodes a protein. This probability is used for the classification and scoring of gene candidates. With large scale training, our method provides fast single fragment predictions with good sensitivity and specificity on artificially fragmented genomic DNA. Additionally, this method is able to predict translation initiation sites accurately and distinguishes complete from incomplete genes with high reliability.

Conclusion: Large scale machine learning methods are well-suited for gene prediction in metagenomic DNA fragments. In particular, the combination of linear discriminants and neural networks is promising and should be considered for integration into metagenomic analysis pipelines. The data sets can be downloaded from the URL provided (see Availability and requirements section).

Background

Communities of natural microorganisms often encompass a bewildering range of physiological, metabolic, and genomic diversity. The microbial diversity in most environments exceeds the biodiversity of plants and animals by orders of magnitude. Phylogenetic surveys of complex ecosystems such as soils and sediments have demonstrated that the multitude of discrete prokaryotic species represented in a single sample goes far beyond the number and phenotypes of known cultured microorganisms [1,2]. Direct cultivation or indirect molecular approaches have been used to explore and to exploit this enormous microbial diversity. Cultivation and isolation of microorganisms are the traditional methods. It has been estimated that less than 1 % of environmental microorganisms are culturable using standard cultivation methods. Thus, only a tiny portion of the gene pool of natural microbial communities has been analyzed so far [2-4].

To circumvent some of the limitations of cultivation approaches, indirect molecular methods, such as metagenomics have been developed. Metagenomics is based on the direct isolation, cloning, and subsequent analysis of microbial DNA from environmental samples without prior cultivation [5-7]. Function- and sequence-based analysis of metagenomic DNA fragments have resulted in the identification of a variety of novel genes and gene products [6,8,9]. In addition, partial sequencing of metagenomes, such as those from the acid mine biofilm (75 Mbp) [10], Minnesota farm soil (100 Mbp) [11], and Sargasso Sea (1,600 Mbp) [12], have provided a better understanding of the structure and genomic potential of microbial communities.

A major goal of metagenomic sequencing projects is the identification of protein coding genes. Most genes in metagenomic fragments are currently identified by homology to known genes by employing other methods, e.g. BLAST [13]. The disadvantage of such an approach is obvious: it is impossible to find novel genes that way. Particularly in cases where metagenomic studies aim to discover new proteins, homology search is an inadequate tool for gene prediction.

The computational ab initio prediction of genes from microbial DNA has a long history, and a number of tools have been developed and employed for gene prediction and annotation of genomic sequences from single prokaryotic species (e.g. GLIMMER [14] and GeneMark.hmm [15]). A minor restriction in the application of some conventional approaches to metagenomes is that they are based on the identification of open reading frames (ORFs), which begin with a start codon and end with an in-frame stop codon. Sequenced metagenomes comprise a collection of numerous short sequencing reads of varying length depending on the employed sequencing technique. A typical metagenomic fragment derived by Sanger sequencing [16] is approximately 700 bp long and contains two or fewer genes. The majority of these genes are incomplete, meaning one or both gene ends extend beyond fragment end(s). Therefore, most ORFs in metagenomic sequencing reads will be overlooked by ORF-based gene finders. A more profound problem is that most gene finders for prokaryotic genomes rely on statistical sequence models that are estimated from the analyzed or a closely related genome. Most metagenomic fragments do not bear sufficient sequence information for building statistical models able to distinguish coding from non-coding ORFs. One might consider to derive models from a complete metagenome but the resulting gene prediction quality in fragments from underrepresented species in the metagenome is questionable.

Up to now, there are three approaches for predicting genes from metagenomic DNA fragments. One of these methods is based on BLAST search, where the search is not only applied against databases of known proteins but also against a library constructed from the metagenomic sample itself [17]. In principle, this computationally expensive approach is able to find novel genes, provided that homologues of these genes are contained in the sample. However, it is not clear whether interesting genes will always be conserved in a metagenomic sample. The first method that was developed for ab initio gene prediction in short and anonymous DNA sequences is a heuristic approach of GeneMark.hmm that derives an adapted monocodon usage model from the GC-content of an input sequence [18].

Another method that was developed for ab initio gene prediction in metagenomic DNA fragments is MetaGene [19]. Similar to GeneMark.hmm, MetaGene employs GC-content specific monocodon and dicodon models for predicting genes. The time-efficient two step gene prediction algorithm first extracts ORFs and scores them on the basis of statistical models estimated from fully sequenced and annotated genomes. Subsequently, a dynamic program calculates the final ORF combination from different scores. Additionally, MetaGene utilizes ORF length, the distance from the annotated start codon to the left-most start codon, and distances to neighboring ORFs. Two separate models were estimated from bacterial and archaeal genomes, respectively. The domain specific models are simultaneously applied to each fragment and the higher scoring model is selected for final gene prediction. Results in randomly sampled fragments from annotated genomes indicate that MetaGene provides a high sensitivity in finding genes in fragmented DNA, while the specificity of the predictions is slightly lower. In addition, the performance

of GeneMark.hmm in 700 bp fragments and for complete genomes was investigated (supplementary table S3 and table 1 of [19]). Comparable performance results were obtained for both methods for both types of input sequences.

Here, we present a novel approach for gene prediction in single fragments, which is based entirely on machine learning techniques. In bioinformatics, state-of-the-art machine learning methods are usually applied to problems where, at most, several thousands of examples exist for training and evaluation. In our application, learning has to be performed on large data sets with millions of examples. This requires the use of a learning architecture that is capable of large-scale training and testing. Here, we propose a combination of neural networks and linear discriminants. While linear discriminants are used for the extraction of features from high-dimensional data which characterize codon usage and potential gene starts, a small neural network is used for non-linear combination of these features with additional information on length and GC-content of gene candidates. Neural networks in combination with linear discriminants or positional weight matrices have also been applied to other gene prediction problems, for instance in promoter recognition [20].

To provide comparability in our experimental evaluation, we use a setup that is similar to the one used for the initial evaluation of MetaGene. We test our program on fragments from thirteen species. However, we provide some important extensions: We use a higher number of fragments which are randomly sampled from the test genomes to avoid any bias that may result from a particular fragmentation technique. The higher number of fragments is used to cope with the variance across different (repeated) sampling experiments. In addition, we provide a detailed analysis of the translation initiation site (TIS) prediction performance and we also investigate the ability to discriminate between complete and incomplete genes.

Methods

Most prokaryotic protein coding genes consist of a start codon, followed by a variable number of consecutive in-frame codons and are terminated by a stop codon. This particular arrangement of codons is commonly referred to as open reading frame (ORF). The sole identification of ORFs is not sufficient for prokaryotic gene prediction because the majority of ORFs in a genome are, in fact, non-coding.

In DNA fragments, ORFs frequently exceed the fragment ends. We therefore extend the ORF definition to *incomplete* ORFs.

The fact that start codons are identical to some *regular* codons results in a high number of related ORFs that share a stop codon but have different start codons. We term such a set of related ORFs an ORF-set and we name the possible start codons of an ORF-set translation initiation site (TIS) candidates. Figure 1 illustrates possible cases of ORF occurrence in a DNA fragment: In case 1, the complete ORF-set is located in the fragment. Additional TIS candidates for this ORF-set can not occur because of an upstream in-frame stop codon. Predicted genes from this ORF-set will always be complete. In case 2, only TIS candidates are located inside the fragment. The range for upstream TIS is again limited by an in-frame stop codon. This candidate, if classified as coding, would result in the prediction of an incomplete gene. In case 3, the stop is located in the fragment. Some TIS candidates are contained in the fragment but there might exist TIS candidates outside the fragment. An ORF-set of this type may result either in a complete or in an incomplete gene. Case 4 is complementary to case 2. Only a stop codon is located inside the fragment. Case 5 and 6 are fragment-spanning ORF-sets, where 5 also includes TIS candidates inside the fragment. Predictions from case 5 will be incomplete but may have a start codon. Case 5 and 6 can both result in the prediction of incomplete genes without start and stop codons.

Our gene prediction algorithm is designed for the discrimination of coding from non-coding ORFs. After the identification of all ORFs in a fragment, we extract features from those ORFs using linear discriminants. Subsequently, we use a neural network that has been particularly trained for the classification of ORFs as coding or non-coding. Classification is based on a gene probability that the neural network assigns to every ORF. Because gene-containing ORF-sets usually comprise of more than one candidate, several ORFs of such an ORF-set may be assigned a high probability by the neural network. The final gene prediction is achieved by a »greedy« method that selects the most probable ORFs that overlap by, at most, 60 bases.

Machine Learning Techniques

To predict whether a particular ORF actually corresponds to a protein coding region or to a non-coding region, we use a neural network for binary ORF classification. In the following sections, we will first describe the features utilized as inputs for the neural network. Subsequently, we will depict the neural network architecture and the methods we used for large scale training and validation from labeled ORFs in artificial fragments.

Features

For realization of the neural network, we use seven features based on sequence characteristics of ORFs. As net-

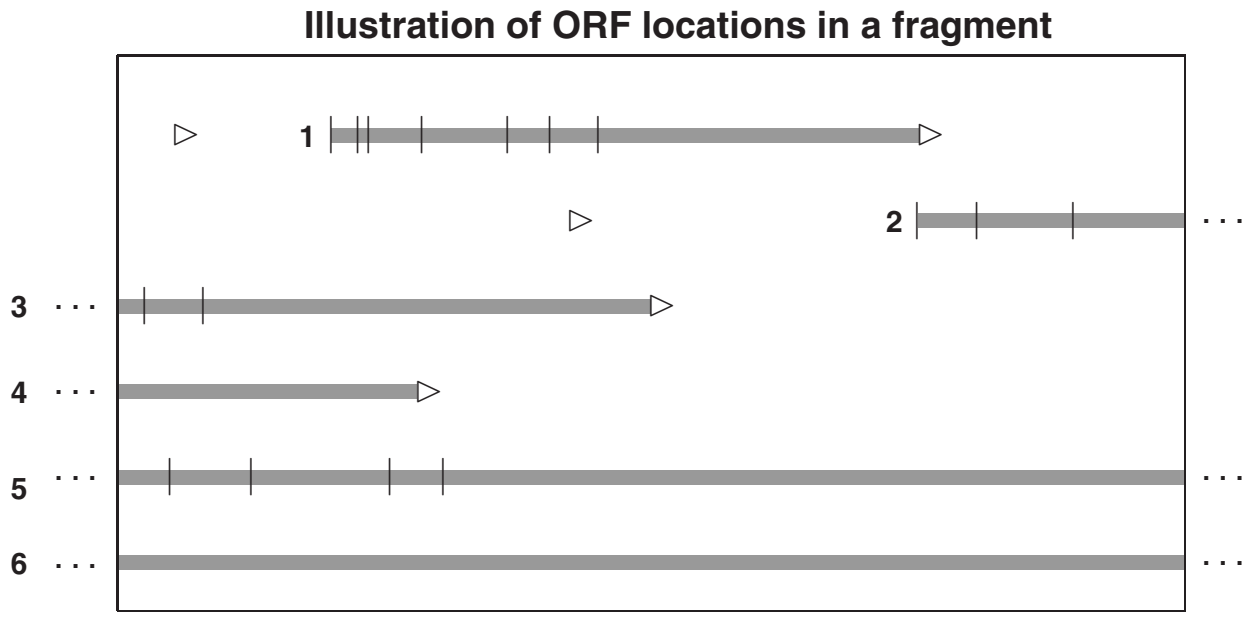


Figure 1
The figure illustrates possible localizations of open reading frames (ORFs) in a fragment (shown only for the forward strand). ORFs are shown as grey bars, »«denotes stop codons, »|« indicates the position of translation initiation site candidates. ORFs that are related by a common stop codon are grouped and we refer to them as ORF-sets. The box symbolizes the fragment range. Everything that might be located outside the box is invisible to gene prediction algorithms. Further explanations are given in section »Methods«.

work inputs, these sequence features are subject to a separate preprocessing step. Below, we explain the methods for computation of these features in detail.

Codon and Dicodon Usage

The perhaps most important features for the discrimination between coding and non-coding ORFs can be derived from codon usage, in particular from 4^3 monocodon and 4^6 dicodon frequencies. These frequencies represent the occurrences of successive trinucleotides (non-overlapping) and hexanucleotides (half-overlapping), respectively. For the characterization of monocodon and dicodon usage, we compute two features based on linear discriminant scores.

Linear discriminants were obtained from training with annotated sequence data. We used coding and non-coding regions from annotated genomes as positive and negative examples, respectively (see section »Training Data for Feature Preprocessing«). Examples are represented by vectors of frequencies of 4^3 and 4^6 possible monocodons and dicodons, respectively. In the following, we describe discriminant training for the monocodon case. The same training procedure was applied to the dicodon case.

For the i-th example, we denote a monocodon frequency vector as $\mathbf{x}_M^i \in \mathbb{R}^{64}$, which is the i-th column of the data matrix $\mathbf{X}_M$, containing all training vectors. To remove length information from these data, all training vectors are normalized to unit Euclidean norm. The corresponding label $\gamma_M^i \in \{-1, 1\}$, which is the i-th element of the label vector $\mathbf{y}_M$, indicates whether the example represents a coding ($\gamma_M^i = 1$) or non-coding ($\gamma_M^i = -1$) region. For training of the discriminant weight vector $\mathbf{w}_M$, we use a regularized least squares approach, i.e. we minimize the following regularized error:

$$E(\mathbf{w}_M) = \frac{1}{n}\sum_{i=1}^{n}(\mathbf{w}_M \cdot \mathbf{x}_M^i - \gamma_i)^2 + \lambda \mathbf{w}_M \cdot \mathbf{w}_M \qquad (1)$$

where » · « denotes the dot product. The minimizer of E is obtained by [21]:

$$\mathbf{w}_M = (\mathbf{X}_M \mathbf{X}_M^T + \lambda\, n\mathbf{I})^{-1} \mathbf{X}_M \mathbf{y}_M \qquad (2)$$

with $d \times d$ identity matrix $\mathbf{I}$ and with upper T and -1 indicating matrix transposition and inversion, respectively.

The computational cost scale linearly with the number of examples, which makes the approach well suited for large scale learning. Doing the same for the dicodon frequency discriminant vector $\mathbf{w}_D$, we obtain two discriminant scores that serve as the first two input features of the neural network:

$$x_1 = \mathbf{w}_M \cdot \mathbf{x}_M, x_2 = \mathbf{w}_D \cdot \mathbf{x}_D. \quad (3)$$

To adjust the regularization parameter λ, we measure the discriminative power of the respective classifier by means of the area under precision recall curve (»auPRC«) as explained in section »Measures of Performance«. Thereby, we choose a $\lambda \in \{10^m | m = -8, -7, ..., 6\}$ to maximize the auPRC on an independent validation set (see section »Training Data for Feature Preprocessing«).

Translation Initiation Site

A discriminant is derived from up- and downstream regions of translation initiation site (TIS) examples. Here we use a 60 basepair (bp) window centered on a potential start codon at window position 31 (see section »Training Data«). We encode the trinucleotide occurrences in that window to yield binary indicator vectors. In each of its 3712 dimensions (64 trinucleotides × 58 positions), a vector indicates whether a certain trinucleotide occurs at a particular window position. Training of the discriminant proceeds in the same way as for the previous two discriminants based on codon usage. Again, we select the regularization parameter $\lambda \in \{10^m | m = -8, -7, ..., 6\}$ by maximization of the auPRC on an independent validation set.

Because not all genes have a potential TIS region we do not use the TIS score $s = \mathbf{w}_T \cdot \mathbf{x}_T$ directly, but instead we take the posterior probabilities of being a TIS or not. For computation of the posterior probabilities, we use Gaussian probability density functions of the score:

$$p(s \mid \mu, \sigma) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{1}{2\sigma^2}(s-\mu)^2\right) \quad (4)$$

where μ stands for mean and σ for standard deviation.

The features x_3 and x_4 were obtained from a mixture of two Gaussians

$$p(s) = \pi^+ p(s|\mu^+, \sigma^+) + \pi^- p(s|\mu^-, \sigma^-) \quad (5)$$

with parameters estimated from scores of positive and negative training examples, respectively (π^+ and π^- are the a priori probabilities of the two classes):

$$x_3 = \frac{\pi^+ p(s \mid \mu^+, \sigma^+)}{p(s)}, \quad x_4 = \frac{\pi^- p(s \mid \mu^-, \sigma^-)}{p(s)}. \quad (6)$$

If no TIS candidate is present, both probabilities are set to zero for that ORF. Note that this case is different from the case of missing values, which can be solved by assigning a priori probabilities for true and false TIS. Here we encounter the possible case where we know that none of the two categories is adequate.

Length features

Another feature for discrimination between coding and non-coding ORFs is the sequence length of the ORF. Here, it is important to distinguish between complete and incomplete ORFs. For incomplete ORFs, the observable »incomplete length« is merely a lower bound for the unobservable »complete length« of that ORF and therefore should be treated in a different way. Consequently, we use one »incomplete« and one »complete length« feature. For a particular ORF, only the feature that corresponds to the type of ORF has non-zero value. The value is simply the observed length divided by the maximal length $l_{\max}$. In our evaluation, we set $l_{\max}$ to 700 bp. In this way we obtain two more features x_5, $x_6 \geq 0$ for complete and incomplete length.

GC-content

As a last feature $x_7 \in [0, 1]$, we use, for each ORF, the GC-content estimated from the whole fragment in which this ORF occurs.

Neural Network

We use standard multilayer perceptrons with one layer of k hidden nodes and with a single logistic output function. Within a binary classification setup with labels $y_i = 1$ (»true«) or $y_i = 0$ (»false«) the output of the neural network can be viewed as an approximation of the posterior probability of the »true« class [22]. In our case, the »true« class represents coding ORFs and therefore the network output can be interpreted in terms of a gene probability. For an input feature vector $\mathbf{x}$, the k hidden layer activations z_i based on input weight vectors $\mathbf{w}_I^i$ and bias parameters b_I^i are

$$z_i = \tanh(\mathbf{w}_I^i \cdot \mathbf{x} + b_I^i). \quad (7)$$

Putting the z_i into a vector $\mathbf{z}$, the output of the network, i.e. its prediction function based on weight vector $\mathbf{w}_O$ and bias b_O, is

$$g(z) = \frac{1}{1+\exp(-w_O \cdot z - b_O)}. \quad (8)$$

Given a training set $\mathbf{x}_1, \ldots, \mathbf{x}_N$ and a network with weight and bias parameters collected in the vector θ, we now write the corresponding network output as $f(\mathbf{x}_1; \theta), \ldots, f(\mathbf{x}_N; \theta)$. With diagonal matrix $\mathbf{A}$ containing the regularization parameters, the training objective is to minimize the regularized error:

$$E(\ ) = \sum_{i=1}^{N} (f(\mathbf{x}_i; \ ) - y_i)^2 + \ ^T\mathbf{A} \ . \quad (9)$$

The diagonal matrix $\mathbf{A} = \text{diag}(\alpha_1, \ldots, \alpha_1, \alpha_2, \ldots, \alpha_2, \alpha_3, \ldots, \alpha_3, \alpha_4)$ of the regularization term involves four hyperparameters $\alpha_1, \alpha_2, \alpha_3, \alpha_4 > 0$ for separate scaling of the parameters $\mathbf{w}_I^i$, b_I^i, $\mathbf{w}_O$, b_O. Note that the regularization term penalizes the squared magnitude of the weights. For the adaptation of hyperparameters, we utilize the evidence framework [23] based on a Gaussian approximation of the posterior distribution of network weights. The evidence-based adaptation of hyperparameters can be incorporated into the network training procedure and does not require additional validation data. For the minimization of (9) with respect to weight and bias parameters, we use a scaled conjugate gradient scheme, as implemented in the NETLAB toolbox [24]. While weight and bias parameters were initialized randomly according to a standard normal distribution, the hyperparameters were initially set to $\alpha_1 = \alpha_2 = \alpha_3 = \alpha_4 = 0.01$. The complete training scheme performs 50 iterations where each iteration comprises 50 gradient steps and two successive hyperparameter adaptation steps.

Final Candidate Selection

Application of the neural network to a certain fragment results in a list of potential gene candidates with a predicted gene probability above 0.5. Most of these predictions are mutually exclusive in terms of overlap. Many predictions even belong to the same ORF-set, differing only in the position of the start codon. In order to obtain a list $\mathcal{G}$ of final genes for a particular fragment, predictions with maximal probability are iteratively selected from the list of candidates $\mathcal{C}$, which is successively reduced according to a maximum overlap constraint. Starting with an empty list $\mathcal{G}$ and an initial list $\mathcal{C}$ containing all fragment-specific ORFs i with gene probability $P_i = f(\mathbf{x}_i; \theta) > 0.5$, we apply the following »greedy« selection scheme:

While $\mathcal{C}$ is nonempty do

- determine $i_{\max} = \arg\max_i P_i$ with respect to all ORFs i in $\mathcal{C}$
- remove ORF $i_{\max}$ from $\mathcal{C}$ and add it to $\mathcal{G}$
- remove all ORFs from $\mathcal{C}$ that overlap with ORF $i_{\max}$ by more than $o_{\max}$ bp

In our evaluation, we set $o_{\max}$ to 60 bp, which corresponds to the minimal gene length we consider for prediction.

Training Data

Our machine learning approach for gene prediction in metagenomic DNA fragments is based on learning the characteristics of coding and non-coding regions from 131 fully sequenced prokaryotic genomes [see Additional file 1] and their GenBank [25] annotation for protein coding genes. The training genomes correspond to the ones that were used for building the statistical models of MetaGene except that we excluded *Pseudomonas aeruginosa* from the training set because a subset of reliably annotated genes that is valuable for the determination of TIS correctness is available for this species. All training and test data sets described in this article are based on the initial extraction of ORFs with a minimal length of 60 bp. Two types of ORFs are distinguishable: *Complete* ORFs begin with a start codon (ATG, CTG, GTG or TTG), and are followed by a flexible number of subsequent codons and conclude with a stop codon (TAG, TGA or TAA). *Incomplete* ORFs stretch from one fragment end to a stop or start codon or to the other fragment end without being interrupted by another in-frame stop codon (compare Figure 1).

In the following paragraphs, we first describe the preparation of training data sets for feature preprocessing and for training of the neural network. Subsequently, we specify the compilation of a test data set for performance evaluation.

Training Data for Feature Preprocessing

Monocodon, dicodon and TIS feature extraction from ORFs require a preprocessing step that is based on the separate training procedure described in section «Codon and Dicodon Usage». Training examples for feature preprocessing were randomly sampled from complete genomes to a coverage of 50 %. Two separate training sets were compiled. For the mono- and dicodon frequencies train-

ing set, DNA sequences of genes defined by their exact start and stop codon position served as positive examples ($\approx 1.9 \times 10^5$). The longest candidate out of each non-coding ORF-set was selected for the composition of negative examples ($\approx 2.8 \times 10^6$).

Training of the TIS discriminant was carried out on symmetric 60 bp sequence windows around start codons. The sequence windows of annotated start codons served as positive examples (1.9×10^5) while the windows around other possible start codons of the same ORF-sets were used as negative examples (5.6×10^6).

The examples for both training data sets were randomly split into 50 % for discriminant training and 50 % for validation of the regularization parameter.

Training Data for the Neural Network

The neural network was trained with the extracted features from ORFs in 700 bp fragments that were randomly excised to a 1-fold genome coverage from each training genome. We define an n-fold coverage as the amount of sampled DNA that is in total length (bp) n times longer than the original genome sequence. Annotated genes in these fragments were used as positive examples for coding regions ($\approx 2.6 \times 10^6$) while one candidate out of each non-coding ORF-set was randomly selected for the negative examples ($\approx 4.5 \times 10^6$). The data sets were randomly split into 50% for neural network training and 50 % for validation of the network size (see section »Neural Network«).

Test Data and Experimental Evaluation

The performance of our gene prediction algorithm was evaluated on artificial DNA fragments from three archaeal and ten bacterial species (see Table 1) whose genera were not used for training. Fragments of the lengths 100 to 2000 bp (in intervals of 100 bp) were randomly sampled from each genome to a 5-fold genome coverage for each length. We used the fragments of all lengths to investigate gene prediction performance of our method, which was trained on fragments with the length 700 bp.

Table 1: Genomes of microbial species that were used for the evaluation of our method. The upper three species are archaea while the lower ten species belong to the bacterial domain. The table shows GenBank accession numbers (GenBank Acc.), and genome sizes (Size).

Species	GenBank Acc.	Size (Mbp)
Archaeoglobus fulgidus	NC_000917	2.2
Methanococcus jannaschii	NC_000909	1.7
Natronomonas pharaonis	NC_007426	2.6
Buchnera aphidicola	NC_002528	0.6
Burkholderia pseudomallei	NC_006350, NC_006351	7.2
Bacillus subtilis	NC_000964	4.2
Corynebacterium jeikeium	NC_007164	2.5
Chlorobium tepidum	NC_002932	2.2
Escherichia coli	NC_000913	4.6
Helicobacter pylori	NC_000921	1.6
Pseudomonas aeruginosa	NC_002516	6.3
Prochlorococcus marinus	NC_007577	1.7
Wolbachia endosymbiont	NC_006833	1.1

A more detailed analysis was carried out on 700 bp fragments (also sampled to a 5-fold coverage), including a comparison to MetaGene. In order to determine statistical significance, we used 10 replicates of each randomly sampled fragment stack.

Gene prediction performance was evaluated by comparing predictions of our method to known annotated genes in fragments. The GenBank annotation for protein coding genes was used to measure general gene prediction performance. However, the GenBank gene start annotation has previously been suspected to be inaccurate [26]. Therefore, we used »reliable gene annotation subsets« [27] for the evaluation of translation initiation site (TIS) prediction performance: all genes with an experimentally verified TIS from »EcoGene« for *Escherichia coli* [28], experimentally verified genes of the *Bacillus subtilis* GenBank annotation (all non-y genes) and the »PseudoCAP« (Pseudomonas community annotation project) annotation of *Pseudomonas aeruginosa* [29].

Measures of Performance

The capability of detecting annotated genes (and genes including their annotated TIS) was measured as sensitivity:

$$\mathrm{Sens} = \frac{\mathrm{TP}}{\mathrm{TP+FN}} \tag{10}$$

For gene prediction sensitivity, $\mathrm{TP}_{\mathrm{gene}}$ (true positives) denotes correct matches and $\mathrm{FN}_{\mathrm{gene}}$ (false negatives) indicate overlooked genes. We counted all predictions as $\mathrm{TP}_{\mathrm{gene}}$ that match at least 60 bp in the same reading-frame to an annotated gene.

In one experiment, we compared gene predictions to a subset of genes that have a reliably annotated gene start in the fragment. For this subset, we measured TIS prediction sensitivity. Here, $\mathrm{TP}_{\mathrm{TIS}}$ are genes with correctly predicted TIS and $\mathrm{FN}_{\mathrm{TIS}}$ are genes whose correct start codons were not predicted.

The reliability of gene predictions was measured by specificity:

$$\text{Spec} = \frac{\text{TP}_{\text{gene}}}{\text{TP}_{\text{gene}} + \text{FP}_{\text{gene}}} \quad (11)$$

Gene prediction specificity was calculated with predicted genes that do not correspond to any gene in the annotation as FP_{gene} (false positives).

To provide a suitable composite measure of sensitivity and specificity we use the harmonic mean, which corresponds to a particular realization of the F-measure [30]:

$$\text{HarmonicMean} = \frac{2 * \text{Sens} * \text{Spec}}{\text{Sens} + \text{Spec}}. \quad (12)$$

To measure the discriminative power of the codon usage and TIS discriminants for feature extraction (see »Features«), we use the area under *precision recall curve* (auPRC). The precision recall curve shows for each possible score threshold the relation of sensitivity (on the x-axes) and specificity (on the y-axes). Sensitivity and specificity are not sufficient for measuring TIS prediction performance. When applied to TIS prediction, these measures rather reflect general gene prediction performance than accuracy of TIS prediction. 'TIS correctness' was therefore measured by the percentage of correctly predicted TIS within a subset of true positive gene predictions TP_{gene} that have an annotated start codon within the fragment (TP_{gene} that have annotated start codon wihin the fragment ($\text{TP}_{\text{gene}}*$):

$$\text{TIS correctness} = \frac{\text{TP}_{\text{TIS}}}{\text{TP}_{\text{gene}^*}}. \quad (13)$$

Accuracy of complete/incomplete gene type prediction was calculated on the basis of correctly predicted genes with an existing true TIS:

$$\text{GeneTypeAccuracy} = \frac{\text{TP}_{\text{complete}} + \text{TN}_{\text{complete}}}{\text{TP}_{\text{gene}}} \quad (14)$$

where $\text{TP}_{\text{complete}}$ and $\text{TN}_{\text{complete}}$ account for the number of genes within TP_{gene} that have correctly been predicted as complete and incomplete.

Results and Discussion

In the following sections, we first describe and discuss the results of discriminant and neural network validation which led to the choice of a hyperparameter λ and a suitable number of nodes for the neural net. Subsequently, we show and discuss gene prediction performance results of the neural network on several fragment lengths and in 700 bp fragments.

Discriminant Validation

Training of the linear discriminants for monocodon, dicodon and TIS features requires the validation of the regularization parameter λ . For each of the three discriminants, we chose λ from the set of values $\{10^m | m = -8, -7, ..., 6\}$ by maximizing the area under precision recall curve (auPRC) on separate validation data. While for the TIS discriminant, a well-defined maximum was achieved for an intermediate $\lambda = 10^{-2}$, for the monocodon and dicodon case the maximum was achieved for the smallest value $\lambda = 10^{-8}$. However, as shown in Additional file 2, for small λ values the auPRC performance in these cases reaches a plateau and therefore we did not try smaller values. The resulting discriminant weights for the 64 monocodons are shown in Additional file 1. The high negative weights for the three stop codons TAA, TAG, TGA are due to the large fraction of negative examples. Because negative examples are, by a factor 10, more frequent than positive examples in the training set, a negative shift of the discriminant score is induced by codons that, like stop codons, are present in any example in any of the two classes.

Network Validation

In principle, the evidence-based hyperparameter adaptation (see section »Neural Network«) obviates the search for an adequate size of the network, i.e. to find a suitable number k of hidden nodes. The network size has just to be large enough to provide maximum performance, while larger nets would automatically be subject to stronger regularization in terms of larger regularization parameters. Nevertheless, network size is crucial in terms of computational cost for training and testing.

In order to find a small network with sufficient performance, we started to train networks of increasing size. Trying networks with k = 5, 10, ..., 25 nodes, we found the performance to reach a nearly flat plateau within that k-range, with only very slight increase above k = 15 [see Additional file 2]. Performance was measured in terms of the harmonic mean criterion (see section »Measures of Performance«), computed on an independent validation set (see section »Training Data«). For the final predictions on the test data, we used the largest network with k = 25 nodes.

While training of neural networks is a time consuming process, computing predictions with a trained network on new data is very fast. In our case, training a network with k = 25 hidden nodes from $\cong 3.6 \times 10^6$ examples took about 190 cpu hours (AMD Opteron, 2 GHz). The training scheme described above was applied in parallel to 5 networks with different (random) initialization of parameters to avoid weak local minima of the regularized error (9). According to the lowest error, the best resulting network was selected for the final predictions within the test

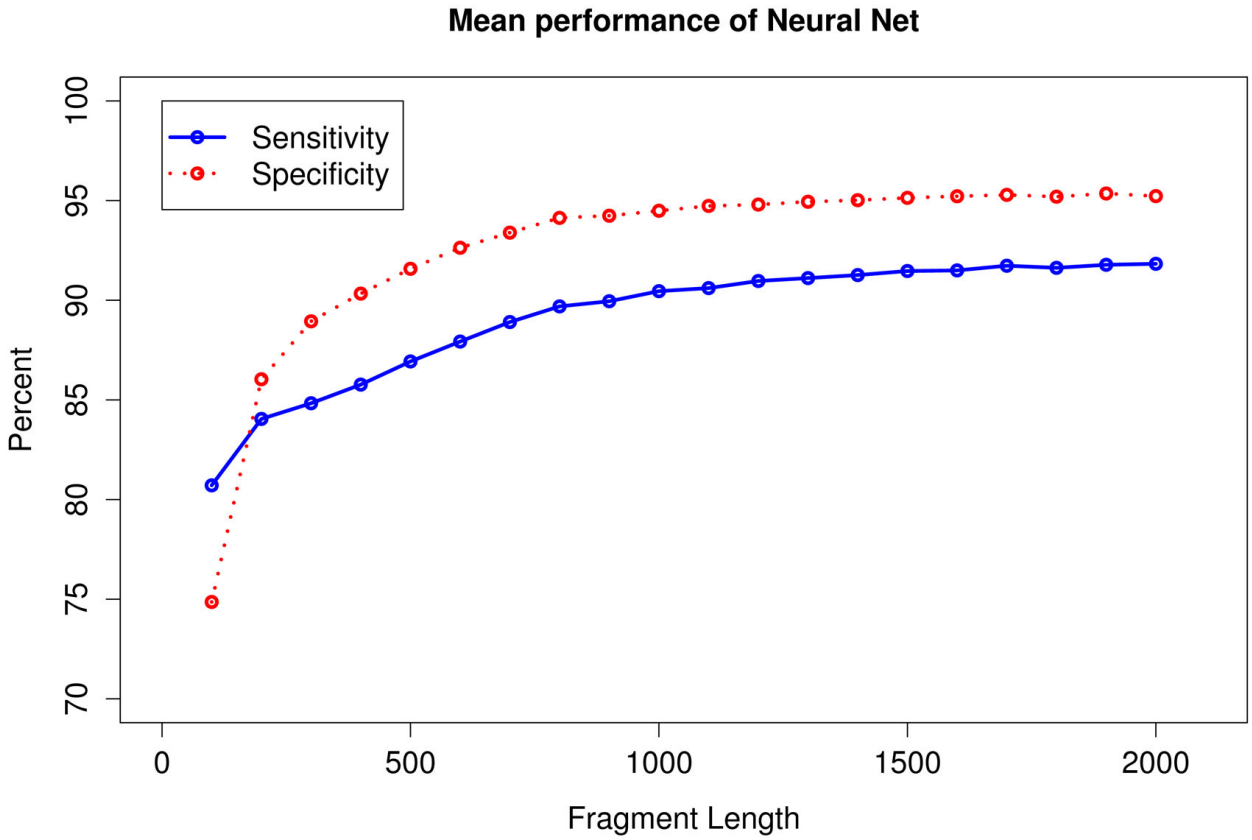


Figure 2
Average gene prediction performance of the neural network in fragments of the lengths 100 to 2000 bp. The performance values from thirteen test species were averaged by arithmetic mean.

setup. In contrast, testing of the same number of examples, i.e. prediction on more than three million candidates, only took ≅ 16.5 seconds on the same machine.

The parameters of the neural network with 25 nodes that we used for further evaluation are given in Additional file 1.

Gene Prediction Performance in DNA fragments

To evaluate the performance of our machine learning approach, we tested the method on artificially fragmented genomes. In the following section, we present the results in general gene prediction performance on various fragment lengths. Subsequently, we analyze gene prediction performance, TIS prediction correctness and complete/incomplete gene type prediction accuracy in detail for fragments of length 700 bp, which corresponds to the fragment length on which the neural network was trained.

Performance in Fragments of Different Lengths

The predictions of our method in DNA fragments with lengths ranging from 100 to 2000 bp from thirteen species were compared to the GenBank [25] annotation for protein coding genes. Note that on average 15 % of 100 bp fragments do not contain any annotated gene matching the 60 bp minimal length criterion (complete or incomplete), for 700 bp fragments, this fraction of fragments accounts 3 %, for 2000 bp fragments 0.8 %. The average percentage of complete genes within all annotated genes in our test fragments is 0 % for 100 bp fragments, 8 % for 700 bp fragments and 40 % for 2000 bp fragments [see Additional file 2]. The mean of gene prediction sensitivity and specificity for all fragment lengths is shown in Figure 2. On 700 bp fragments, our method has an average gene prediction sensitivity of 89 % and an average specificity of 93 %. Sensitivity and specificity slightly increase with growing fragment size. This can be explained by the fact that ORFs carrying distinct mono-/dicodon and TIS signals occur more often in longer fragments. Gene predic-

tion performance decreases with length and sharply drops for fragments shorter than 200 bp.

Before application of our method on real metagenomes, the neural network should be trained and thoroughly evaluated on fragments of a length that corresponds to the real fragments of interest. Metagenomic sequencing projects differ in their aims and in the applied sequencing and annotation strategies. Improved Sanger sequencing from one or both vector insert ends is applied in many metagenomic projects (examples are listed in review [31]) and yields sequencing reads roughly ranging from 500 to 1000 bp. Based on the current results, our method might be particularly useful for improving gene annotation and discovery on Sanger sequencing reads. However, pyrosequencing [32] has also been introduced to metagenomics [33]. The pyrosequencing approach does not involve any cloning step. With recent improvements, pyrosequencing now yields a read length between 200 and 300 bp [34]. In principle, it should be possible to predict genes in such short fragments with our fragment-based techniques but environmental pyrosequencing projects may rather be focused on phylogenetic studies and habitat comparison than on the discovery of new genes. In some metagenomic sequencing projects, long metagenomic inserts (up to 40 kbp) are fully sequenced [35]. Although gene prediction performance of the neural network does not decrease on longer fragments [see Additional file 2], other methods like MetaGene [19] or GeneMark.hmm [18], which also consider the context of putative genes (e.g. operons), may be more suitable for fragments of this size.

Performance in 700 bp Fragments

Predicted genes in fragments with a length of 700 bp from three archaeal and ten bacterial species were compared to the GenBank annotation for protein coding genes. The mean and standard deviation for sensitivity, specificity and the harmonic mean of 10 repetitions per species are shown in Table 2. The neural network has high sensitivity (ranging from 82 to 92 %) and specificity (ranging from 85 to 97 %) in fragments from all species. We could not observe a major performance difference for sensitivity and specificity between archaeal and bacterial fragments but the variation between different species in general is large.

In comparison to MetaGene, the neural network has a higher specificity in fragments from all test species (on average 4.6 % higher). On the other hand, MetaGene has a higher sensitivity in fragments from most species (on average 3.8 % higher). The neural network only shows a higher sensitivity in *Bacillus subtilis* and *Helicobacter pylori*. The overall performance of both methods calculated by harmonic mean is very similar. For some species, the neural network yields a better overall gene prediction performance while MetaGene performs better on other species. In particular, MetaGene performs better in all tested archaea. All local pairwise differences in sensitivity, specificity and harmonic mean between the neural network and MetaGene are significant to a confidence level of 95 % according to Wilcoxon's signed rank test [36] (R-package *exactRankTests* [37]).

A precise TIS prediction is very important in metagenomics since the aim of many environmental sequencing projects is the identification and subsequent experimental investigation of novel genes. For example, the expression of a metagenomic protein in a host organism may fail or yield incorrect results if the predicted start codon is incorrect. Accurate TIS prediction is a difficult task, even for conventional gene finders on complete genomes [38-42]. This is because ATG, CTG, GTG and TTG also occur inside genes.

Table 2: Mean and standard deviation for gene prediction performance of our method (Neural Net) and MetaGene. Performance was measured on 700 bp fragments that were randomly excised from each test genome to 5-fold coverage (ten replications per species). The harmonic mean is a measure that combines sensitivity and specificity.

	SENSITIVITY		SPECIFICITY		HARMONIC MEAN	
Species	Neural Net	MetaGene	Neural Net	MetaGene	Neural Net	MetaGene
Archaeoglobus fulgidus	87.2 ± 0.21	**93.7 ± 0.15**	**93.4 ± 0.16**	92.7 ± 0.16	90.2 ± 0.17	**93.2 ± 0.14**
Methanococcus jannaschii	91.7 ± 0.17	**95.8 ± 0.14**	**96.2 ± 0.13**	92.7 ± 0.19	93.9 ± 0.10	**94.3 ± 0.15**
Natronomonas pharaonis	87.9 ± 0.22	**95.1 ± 0.09**	**93.9 ± 0.10**	92.7 ± 0.17	90.8 ± 0.16	**93.9 ± 0.12**
Buchnera aphidicola	90.6 ± 0.37	**96.7 ± 0.24**	**95.3 ± 0.31**	91.1 ± 0.29	92.9 ± 0.28	**93.8 ± 0.21**
Burkholderia pseudomallei	87.9 ± 0.11	**94.1 ± 0.11**	**90.1 ± 0.09**	85.1 ± 0.13	89.0± 0.08	**89.4 ± 0.10**
Bacillus subtilis	**91.4 ± 0.16**	89.8 ± 0.14	**95.3 ± 0.09**	89.3 ± 0.19	**93.3 ± 0.10**	89.5 ± 0.14
Corynebacterium jeikeium	89.7 ± 0.24	**91.9 ± 0.12**	**93.8 ± 0.19**	89.2 ± 0.21	**91.7 ± 0.19**	90.5 ± 0.13
Chlorobium tepidum	82.1 ± 0.25	**85.7 ± 0.27**	**91.2 ± 0.17**	88.4 ± 0.26	86.4 ± 0.19	**87.0 ± 0.22**
Escherichia coli	91.7 ± 0.16	**93.3 ± 0.07**	**95.3 ± 0.09**	90.9 ± 0.10	**93.5 ± 0.12**	92.1 ± 0.07
Helicobacter pylori	**92.1 ± 0.11**	90.2 ± 0.14	**96.6 ± 0.15**	89.6 ± 0.23	**94.3 ± 0.11**	89.9 ± 0.15
Pseudomonas aeruginosa	90.4 ± 0.14	**96.2 ± 0.07**	**92.5 ± 0.11**	91.4 ± 0.09	91.4 ± 0.12	**93.7 ± 0.07**
Prochlorococcus marinus	87.2 ± 0.21	**93.7 ± 0.25**	**95.9 ± 0.14**	90.8 ± 0.20	91.4 ± 0.15	**92.2 ± 0.19**
Wolbachia endosymbiont	87.2 ± 0.27	**90.6 ± 0.42**	**85.2 ± 0.44**	71.2 ± 0.54	**86.2 ± 0.29**	79.7 ± 0.45

Table 3: Translation initiation site prediction correctness (TIS correctness) and complete/incomplete classifi-cation accuracy (Gene Type Accuracy) of the Neural Net and MetaGene according to GenBank annotation. Performance was measured on 700 bp fragments that were randomly excised from each test genome to 5-fold coverage (mean and standard deviation for 10 replicates per species are given).

	TIS CORRECTNESS		GENE TYPE ACCURACY	
Species	Neural Net	MetaGene	Neural Net	MetaGene
Archaeoglobus fulgidus	69.8 ± 0.32	73.6 ± 0.32	98.1 ± 0.05	97.2 ± 0.07
Methanococcus jannaschii	69.4 ± 0.52	73.3 ± 0.52	99.0 ± 0.09	97.6 ± 0.12
Natronomonas pharaonis	75.2 ± 0.58	82.9 ± 0.28	96.9 ± 0.16	97.6 ± 0.09
Buchnera aphidicola	86.5 ± 0.40	88.6 ± 0.64	99.1 ± 0.09	98.3 ± 0.21
Burkholderia pseudomallei	70.1 ± 0.45	73.0 ± 0.28	97.6 ± 0.08	96.9 ± 0.09
Bacillus subtilis	79.7 ± 0.32	66.1 ± 0.42	98.6 ± 0.05	97.0 ± 0.08
Corynebacterium jeikeium	78.2 ± 0.49	73.4 ± 0.68	98.1 ± 0.08	96.6 ± 0.11
Chlorobium tepidum	68.1 ± 0.46	71.9 ± 0.45	98.1 ± 0.08	96.7 ± 0.13
Echerichia coli	84.5 ± 0.31	78.2 ± 0.15	98.7 ± 0.06	97.0 ± 0.08
Helicobacter pylori	87.3 ± 0.40	77.1 ± 0.33	99.2 ± 0.09	96.4 ± 0.16
Pseudomonas aeruginosa	78.4 ± 0.22	81.0 ± 0.36	97.7 ± 0.03	97.2 ± 0.07
Prochlorococcus marinus	86.6 ± 0.40	88.6 ± 0.47	99.0 ± 0.07	97.8 ± 0.10
Wolbachia endosymbiont	79.3 ± 0.77	79.9 ± 0.42	98.7 ± 0.13	96.9 ± 0.17

One gene may for example contain several ATGs but only one corresponds to a TIS. Our approach includes a TIS-model that is based on a linear discriminant. We measured TIS prediction performance of our algorithm for all correctly predicted genes that have annotated start codons within a fragment. First, we investigated TIS performance on our complete set of test species fragments according to GenBank annotation. The results are shown in Table 3. TIS correctness of our algorithm varies remarkably between different test species. On some bacterial species, our algorithm reaches a TIS correctness of 87 % (e.g. in *Helicobacter pylori*). The lowest TIS performance can be observed in fragments from the bacterium *Chlorobium tepidum* (68 %). The average TIS correctness of our algorithm is around 78 %. In comparison to this, the highest performance of MetaGene can be observed for fragments of *Prochlorococcus marinus* (89 %), the lowest for fragments of *Bacillus subtilis* (66 %). Note that TIS correctness depends on the number of correctly predicted genes with an annotated TIS. Therefore, TIS correctness of our algorithm is not directly comparable to the one obtained by MetaGene, which detects a higher number of genes. However, the variation in TIS correctness of both methods is large.

A reason for this variation might be that the GenBank gene annotation contains many hypothetical and not experimentally verified genes [26]. Therefore, we also evaluated TIS prediction performance on »reliable annotation subsets« of the bacteria *Escherichia coli*, *Bacillus subtilis* and *Pseudomonas aeruginosa* (see section »Test Data and Experimental Evaluation«). Evaluating gene prediction performance in fragments according to these annotation subsets, our algorithm achieves a highly consistent TIS prediction performance between 81 and 87 % in fragments from all three test species. The TIS prediction sensitivity varies from 68 % to 80 % (see Table 4). In comparison, MetaGene's TIS performance shows a higher variation, ranging from 70 to 84 % while the TIS sensitivity ranges from 62 to 80 %.

The nature of fragmented DNA results in the occurrence of complete and incomplete genes. A gene may be incorrectly predicted as complete or incomplete if it has several TIS candidates of which at least one is located outside the

Table 4: Translation initiation site prediction performance of the new gene prediction algorithm (Neural Net) and MetaGene according to »reliable annotation subsets« (A subset of »verified genes« from »EcoGene« for *Escherichia coli* [28], all non-y genes of the *Bacillus subtilis* GenBank annotation and the »PseudoCAP« annotation of *Pseudomonas aeruginosa* [29]). TIS prediction sensitivity and correctness were measured on artificial 700 bp fragments that were randomly excised from each test genome to 5-fold coverage. Mean and standard deviation over 10 replicates per species are shown.

	SENSITIVITY TIS		TIS CORRECTNESS	
Species	Neural Net	MetaGene	Neural Net	MetaGene
Bacillus subtilis	**73.4 ± 1.79**	62.1 ± 1.43	84.1 ± 0.51	70.2 ± 0.64
Escherichia coli	**80.0 ± 0.68**	75.1 ± 0.61	86.6 ± 0.57	77.5 ± 0.67
Pseudomonas aeruginosa	68.0 ± 0.22	**79.7 ± 0.44**	80.7 ± 0.20	83.7 ± 0.36

fragment. Due to the short fragment length of 700 bp, the vast majority of annotated genes (≈90 %) in our test fragments is incomplete. The experimental strategy of many metagenomic projects relies on sequencing the fragment from one or both ends of the vector insert. Although the insert is not always sequenced completely, sequencing of the entire fragment is possible in case the biologist is interested in further analysis of an incomplete gene. Therefore, it is important to know whether a gene is contained in a sequencing read completely or incompletely.

We evaluated the percentage of genes that were correctly classified as complete or incomplete within the correctly identified genes according to GenBank annotation. Our method achieves an average accuracy of 98 % with little variation (see Table 3). It can be noted that MetaGene slightly more often misclassifies genes concerning their completeness. Note here that the performance indices of the neural network and MetaGene in Table 3 are not directly comparable because they rely on different numbers of correctly identified genes.

Remarks on the Experimental Setup

The evaluation of computational methods for metagenomic gene prediction is troubled by the fact that reliably annotated metagenomes are not available. Some metagenomes have been subject to annotation for several years by now, but their gene annotation is far from complete. Particularly, the exact location of gene starts on metagenomes has been verified experimentally only in rare cases. Currently, the only way to reliably investigate gene prediction accuracy is the evaluation on DNA fragments from complete microbial genomes. For the evaluation of our method, we used an experimental setup similar to the one proposed by the authors of MetaGene in order to keep both methods comparable. MetaGene relies on statistical models built from 116 bacterial and 15 archaeal genomes. These species were selected to represent every genus from GenBank in the year 2006. By now, species belonging to many additional genera have been fully sequenced and annotated. Members of these genera should be included in the training set of a future gene prediction tool version in order to collect as much information about the characteristics of coding and non-coding ORFs as possible.

It remains an open question, which criteria are most suitable for the selection of training species. In general, taxonomy does not reflect phylogeny properly. Some species of different genera for example exhibit highly similar codon usage patterns. Particularly for the identification of novel genes in metagenomes whose biological diversity is yet unknown, the transfer of the GenBank bias toward single species should be avoided in the training data set. To reduce this bias, training genomes could be selected according to other criteria, e.g. GC-content, oligonucleotide frequencies or monocodon/dicodon frequencies in protein coding regions.

The experimental setup chosen here also differs from real metagenomes with respect to sequencing errors. The effect of sequencing errors in terms of base-changes on gene prediction performance of our method would depend on the frequency of such kind of error. The effect of a usually small number of base-errors (less than one error per 10 000 bp after routine fragment end removal [19]) can be neglected. As for other alignment-free methods, like MetaGene, our method is susceptible to frame shifts. Only certain alignment-based methods can be expected to be more robust with regard to this kind of error [17].

Conclusion

Large scale machine learning is well suitable for gene prediction in metagenomic DNA fragments. Due to performance results obtained with the current experimental setup, we suggest that our machine learning approach, with its high gene prediction specificity, TIS correctness and complete/incomplete prediction capabilities, complements MetaGene with its high gene finding sensitivity well. Thus, a combination of both methods should be considered.

Software availability

Linear discriminants and the trained neural network are available as MATLAB files for download at [43]. A command line tool for gene prediction in DNA fragments (Linux, 64-bit architecture) is available from the authors on request.

Availability and requirements

Orphelia: http://orphelia.gobics.de/datasets/

Authors' contributions

KJH developed and implemented the combination of ORF extraction and machine learning modules, assembled training and test data sets and performed the evaluation of the new algorithm and MetaGene. MT developed and implemented ORF-set identification and extraction. KJH and MT contributed biological expertise, T.L. implemented fast versions of the discriminant scoring. RD contributed biological expertise on metagenomics and drafted parts of the manuscript. PM contributed machine learning expertise and designed and implemented the machine learning architecture. BM and TL supported the project and contributed conceptually. KJH and PM wrote the manuscript. All authors read and approved the manuscript.

Additional material

Additional file 1

Tables with training genomes, discriminant weights, and network parameters. *The tables list all genomes that were used for training the neural network (1), present the discriminant weights that were learned for all monocodons (2), and give neural network parameters (3, 4).*
Click here for file
[http://www.biomedcentral.com/content/supplementary/1471-2105-9-217-S1.pdf]

Additional file 2

Supplementary figures. *The figures show the area under precision recall curve for discriminant validation using different λ values (1), the neural network performance with increasing numbers of nodes (2), the percentage of complete genes within all annotated genes per fragment for different fragment lengths (3), and gene prediction performance on fragments ranging from 5000 to 60000 bp (4).*
Click here for file
[http://www.biomedcentral.com/content/supplementary/1471-2105-9-217-S2.pdf]

Acknowledgements

We thank Dr. Heiko Liesegang for supporting the development of our method. We are grateful to Andrew Woehler for proofreading the final version of our manuscript. K.J.H was supported by a Georg-Christoph-Lichtenberg stipend granted by the state of Lower Saxony. T.L. was supported by BMBF project MediGrid (01AK803G).

References

1. Hugenholtz P: **Exploring prokaryotic diversity in the genomic era.** *Genome Biol* 2002, **3(2):**reviews0003.1-0003.8.
2. Torsvik V, ∃vreås L: **Microbial diversity and function in soil: from genes to ecosystems.** *Curr Opin Microbiol* 2002, **5:**240-245.
3. Amann R, Ludwig W, Schleifer K: **Phylogenetic identification and in situ detection of individual microbial cells without cultivation.** *Microbiol Rev* 1995, **59:**143-169.
4. Rappe MS, Giovannoni SL: **The uncultured microbial majority.** *Annu Rev Microbiol* 2003, **57:**369-394.
5. Riesenfeld CS, Schloss PD, Handelsman J: **Metagenomics: Genomic analysis of microbial communities.** *Annu Rev Genet* 2004, **38:**525-552.
6. Handelsman J: **Metagenomics: Application of genomics to uncultured microorganisms.** *Microbiol Mol Biol Rev* 2004, **64(4):**669-685.
7. Daniel R: **The metagenomics of soil.** *Nature Rev Microbiol* 2005, **3:**470-478.
8. Daniel R: **The soil metagenome – a rich resource for the discovery of novel natural products.** *Curr Opin Biotechnol* 2004, **15:**199-204.
9. Streit W, Daniel R, Jaeger KE: **Prospecting for biocatalysts and drugs in the genomes of non-cultured microorganisms.** *Curr Opin Biotechnol* 2004, **15:**285-290.
10. Tyson GW, Chapman J, Hugenholtz P, Allen EE, Ram RJ, Richardson PM, Solovyev VV, Rubin EM, Rokhsar DS, Banfield JF: **Community structure and metabolism through reconstruction of microbial genomes from the environment.** *Nature* 2004, **428:**37-43.
11. Tringe SG, von Mering C, Kobayashi A, Salamov AA, Chen K, Chang HW, Podar M, Short JM, Mathur EJ, Detter JC, Bork P, Hugenholtz P, Rubin EM: **Comparative metagenomics of microbial communities.** *Science* 2005, **308:**554-557.
12. Venter JC, Remington K, Heidelberg JF, Halpern AL, Rusch D, Eisen JA, Wu D, Paulsen I, Nelson KE, Nelson W, Fouts DE, Levy S, Knap AH, Lomas MW, Nealson K, Peterson OWJ, Hoffman J, Parsons R, Baden-Tillson H, Pfannkoch C, Rogers YH, Smith HO: **Environmental shotgun sequencing of the Sargasso Sea.** *Science* 2004, **304:**66-74.
13. Altschul S, Gish W, Miller W, Myers E, Lipman D: **Basic local alignment search tool.** *J Mol Biol* 1990, **215:**403-410.
14. Delcher AL, Harmon D, Kasif S, White O, Salzberg SL: **Improved microbial gene identification with GLIMMER.** *Nucleic Acids Res* 1999, **27(23):**4636-4641.
15. Lukashin A, Borodovsky M: **GeneMark.hmm: new solutions for gene finding.** *Nucleic Acids Res* 1998, **26(4):**1107-1115.
16. Sanger F, Nicklen S, Coulson AR: **DNA sequencing with chain-terminating inhibitors.** *Proc Natl Acad Sci USA* 1977, **74(12):**5463-5467.
17. Krause L, Diaz NN, Bartels D, Edwards RA, Pühler A, Rohwer F, Meyer F, Stoye J: **Finding novel genes in bacterial communities isolated from the environment.** *Bioinformatics* 2006, **22(14):**e281-e289.
18. Besemer J, Borodovsky M: **Heuristic approach to deriving models for gene finding.** *Nucleic Acids Res* 1999, **27(19):**3911-3920.
19. Noguchi H, Park J, Takagi T: **MetaGene: prokaryotic gene finding from environmental shotgun sequences.** *Nucleic Acids Res* 2006, **34(19):**5623-5630.
20. Bajic VB, Seah SH, Chong A, Zhang G, Koh JLY, Brusic V: **Dragon Promoter Finder: recognition of vertebrate RNA polymerase II promoters.** *Bioinformatics* 2002, **18:**198-199.
21. Hastie T, Tibshirani R, Friedman J: *The Elements of Statistical Learning* Berlin: Springer-Verlag; 2001.
22. Bishop CM: *Neural Networks for Pattern Recognition* Oxford: Clarendon Press; 1995.
23. MacKay DJC: **A Practical Bayesian Framework for Backpropagation Networks.** *Neural Comput* 1992, **4(3):**448-472.
24. Nabney IT: *Netlab: Algorithms for Pattern Recognition* New York: Springer-Verlag; 2001.
25. Benson DA, Karsch-Mizrachi I, Lipman DJ, Ortell J, Wheeler DL: **GenBank.** *Nucleic Acids Res* 2007, **35:**D21-D25.
26. Nielson P, Krogh A: **Large-scale prokaryotic gene prediction and comparison to genome annotation.** *Bioinformatics* 2005, **21(24):**4322-4329.
27. Tech M, Meinicke P: **An unsupervised classification scheme for improving predictions of prokaryotic TIS.** *BMC Bioinformatics* 2006, **7(121):**.
28. Rudd KE: **EcoGene: a genome sequence database for *Escherichia coli* K-12.** *Nucleic Acids Res* 2000, **28:**60-64.
29. **PseudoCAP *Pseudomonas aeruginosa* Community Annotation Project** [http://pseudomonas.com/]
30. van Rijsbergen CJ: *Information retrieval* 2nd edition. London: Butterworths; 1979.
31. Chen K, Pachter L: **Bioinformatics for whole-genome shotgun sequences of microbial communities.** *PloS Comput Biol* 2005, **1(2):**106-112.
32. Ronaghi M, Uhlén M, Nyreén P: **A sequencing method based on real-time pyrophosphate.** *Science* 1998, **281(5375):**363-365.
33. Edwards RA, Rodriguez-Britol B, Wegley L, Haynes M, Breitbart M, Peterson DM, Saar MO, Alexander S, Alexander EC, Rohwer F: **Using pyrosequencing to shed light on deep mine microbial ecology.** *BMC Genomics* 2006, **7:**57.
34. Jarvie T, Harkins T: **Metagenomics Analysis Using the Genome Sequencer FLX System.** *Biochemica* 2007, **3:**4-6.
35. Voget S, Leggewie C, Uesbeck A, Raasch C, Jaeger KE, Streit WR: **Prospecting for Novel Biocatalysts in a Soil Metagenome.** *Appl Env Microbiol* 2003, **69(10):**6235-6242.
36. Wilcoxon F: **Individual comparisons by ranking methods.** *Biometrics* 1945, **1(6):**80-83.
37. R Development Core Team: *R: A language and environment for statistical computing* R Foundation for Statistical Computing, Vienna, Austria; 2004. ISBN 3-900051-00-3
38. Suzek BE, Ermolaeva MD, Schreiber M, Salzberg SL: **A probabilistic method for identifying start codons in bacterial genomes.** *Bioinformatics* 2001, **17(12):**1123-1130.
39. Frishman D, Mironov A, Gelfand M: **Starts of bacterial genes: estimating the reliability of computer predictions.** *Gene* 1999, **234(2):**257-265.
40. Ou HY, Guo FB, Zhang CT: **GS-Finder: a program to find bacterial gene start sites with a self-training method.** *Int J Biochem Cell Biol* 2004, **36(3):**535-544.

41. Tech M, Pfeifer N, Morgenstern B, Meinicke P: **TICO: a tool for improving predictions of prokaryotic translation initiation sites.** *Bioinformatics* 2005, **17(21):**3568-3569.
42. Tech M, Morgenstern B, Meinicke P: **TICO: a tool for post-processing the predictions of prokaryotic translation initiation sites.** *Nucleic Acids Res* 2006, **34:**588-590.
43. **sets D** [http://orphelia.gobics.de]

Chapter 4

Orphelia: Predicting Genes in Metagenomic Sequencing Reads

Citation

K. J. Hoff, T. Lingner, P. Meinicke, M. Tech
Orphelia: predicting genes in metagenomic sequencing reads
Nucleic Acids Research 2009 37:W101-W105
doi:10.1093/nar/gkp327

Original Contribution

Assembly of training data for fragment-length specific neural networks; design of figure 1; adaptation of sourcecode that was used for Hoff *et al.* (2008) to the requirements of a user friendly web server application; evaluation of all gene prediction tools, resulting in table 1 and figure 3; manuscript writing (except for the section Implementation and the part in section Methods that concerns the translation initiation site model).

Nucleic Acids Research Advance Access published May 8, 2009

Nucleic Acids Research, 2009, 1–5
doi:10.1093/nar/gkp327

Orphelia: predicting genes in metagenomic sequencing reads

Katharina J. Hoff[1,*], Thomas Lingner[1,2], Peter Meinicke[1] and Maike Tech[1]

[1]Abteilung Bioinformatik, Institut für Mikrobiologie und Genetik, Georg-August-Universität Göttingen, Goldschmidtstr. 1, 37077 Göttingen, Germany and [2]Center for Genomic Regulation, Comparative Bioinformatics Research Group, Biomedical Research Park, c/Dr. Aiguader 88, 08003 Barcelona, Spain

Received February 13, 2009; Revised and Accepted April 20, 2009

ABSTRACT

Metagenomic sequencing projects yield numerous sequencing reads of a diverse range of uncultivated and mostly yet unknown microorganisms. In many cases, these sequencing reads cannot be assembled into longer contigs. Thus, gene prediction tools that were originally developed for whole-genome analysis are not suitable for processing metagenomes. Orphelia is a program for predicting genes in short DNA sequences that is available through a web server application (http://orphelia.gobics.de). Orphelia utilizes prediction models that were created with machine learning techniques on the basis of a wide range of annotated genomes. In contrast to other methods for metagenomic gene prediction, Orphelia has fragment length-specific prediction models for the two most popular sequencing techniques in metagenomics, chain termination sequencing and pyrosequencing. These models ensure highly specific gene predictions.

INTRODUCTION

Metagenomics is an approach to the characterization of microbial genomes without the cultivation of individual species under laboratory conditions. In metagenomic sequencing projects, DNA is directly isolated from the environment and sequenced. Currently, the most common sequencing methods utilized in this field are chain termination sequencing (also named Sanger sequencing) (1), which yields an average read length of ~700 bp, and the more cost efficient pyrosequencing (2), which results in reads of average length ~300 bp. Regardless of the read length, it is in many cases impossible to reliably assemble metagenomic sequencing reads into longer contigs because diversity in metagenomic samples is often too large to provide a high sequencing coverage of single species. To answer one of the major questions of metagenomic sequencing projects, which parts of the sequencing reads encode for proteins, methods are required that can identify genes directly in short and anonymous DNA fragments.

In principle, metagenomic gene prediction is accomplished by two approaches. One is the identification of genes through homology-based methods, for instance by BLAST search of the input sequence against a database of known proteins (3). This approach is limited to the prediction of genes that are highly similar to already known genes. By the clustering of open reading frames (ORFs), homology-based methods can also find novel genes which are conserved within the metagenomic sample (4,5). However, these methods become computationally expensive for large samples. A different approach is gene prediction by means of statistical models. Model-based gene prediction methods have the advantage that they can discover novel genes at lower computational cost and without the prerequisite of a high conservation of these genes within the sample. On the other hand, most model-based methods are sensitive to sequencing errors in form of frame shifts. Up to now, three model-based gene prediction tools for metagenomic DNA fragments are available, namely MetaGene (6), its successor, the MetaGeneAnnotator (7), and GeneMark with a heuristic model (8). All three tools are available as web server applications. In contrast to the MetaGene and MetaGeneAnnotator web servers, the GeneMark web server was not designed to treat single entries of a multiple fasta file separately, which limits its applicability to metagenomic data. Nevertheless, all tools achieve a good performance on fragments of Sanger read length. Prediction accuracies on 300 bp DNA fragments are lower.

Here, we introduce the *ab initio* gene prediction web server application 'Orphelia', which is based on our previously published machine learning approach to metagenomic gene prediction (9). While the other three tools utilize the same prediction model for all read lengths, Orphelia currently supports two separate models for the most common sequencing techniques in metagenomics, thereby also providing highly specific gene predictions in fragments <300 bp. A high gene prediction specificity can

*To whom correspondence should be addressed: Tel: +49 551 391 3884; Fax +49 551 391 4929; Email: katharina@gobics.de

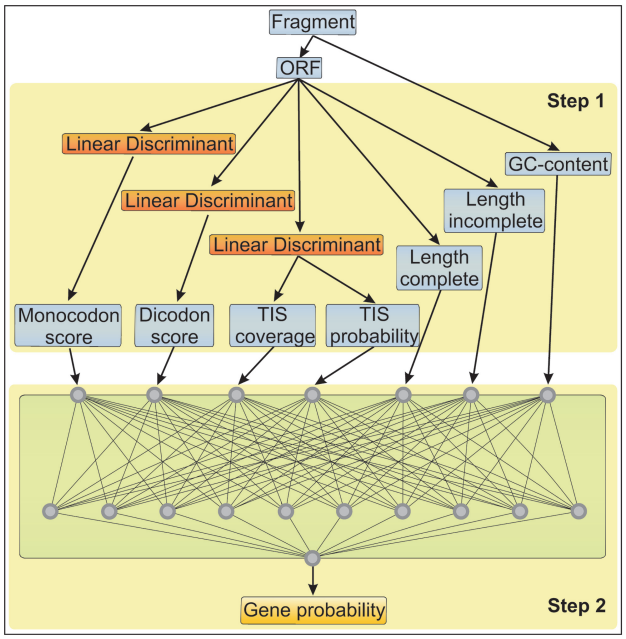


Figure 1. Orphelia's ORF scoring model. In Step 1, 7 ORF/fragment features are computed. Step 2 calculates a final gene probability, combining the features by means of a neural network.

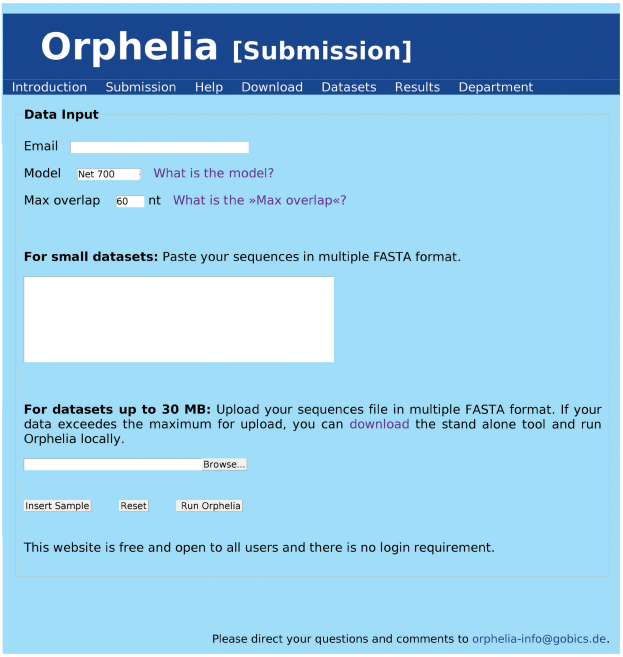


Figure 2. Screenshot of the Orphelia web server application submission page.

be very important for high-throughput metagenome analysis, because the large number of sequences usually makes a manual curation of the predictions impossible.

METHODS

In a first step, Orphelia identifies all ORFs in the input sequence. By our definition, ORFs begin with a start codon (ATG, CTG, GTG, or TTG), are followed by at least 18 subsequent triplets, and end with a stop codon (TGA, TAG, or TAA). Due to the short input sequence length, we also consider incomplete ORFs of at least 60 bp input length that lack a start and/or stop codon. After extraction, all ORFs are scored by a gene prediction model that is based on machine learning techniques. Finally, a greedy method with a maximal overlap constraint selects a combination of highly probable genes.

The gene prediction model is sketched in Figure 1. At first, features for monocodon usage, dicodon usage and translation initiation sites are extracted from the ORF sequence using linear discriminants. The discriminants were trained on 131 fully sequenced prokaryotic genomes (9). After feature extraction, an artificial neural network combines the sequence features with ORF length and fragment GC-content, and computes a posterior probability of an ORF to encode a protein. The neural network was trained on randomly excised DNA fragments of a specified length from the genomes that were used for linear discriminant training. In our previous publication, we provided a prediction model in which the neural network was trained on 700 bp fragments for predicting genes in Sanger read length fragments. We showed that this model is robust with respect to varying sequence length (above ~300 bp). On fragments as short as ~300 bp, we observed a drastic decrease in performance. Therefore, the Orphelia web server also provides an additional prediction model that was trained on 300 bp fragments, which corresponds to the average read length of pyrosequencing.

Besides the discriminant-based translation initiation site (TIS) probability as inferred from a 60 bp TIS region around the potential start codon, we now use the 'TIS coverage' as an additional feature. The TIS coverage is the fraction of the TIS region, which is actually contained in the sequence fragment. This feature accounts for incomplete TIS regions and completely missing start codons, which imply a zero coverage.

WEB SERVER

Input

The Orphelia submission page is shown in Figure 2. Orphelia requires as input data a set of DNA sequences in standard multiple FASTA format. Small data sets can be pasted into the sequence window, larger data sets should be uploaded via the 'Browse' button. Currently, the upload is limited to 30 MB. If a data set exceeds this size, we recommend either the splitting into smaller files, or the usage of our standalone command-line tool for 64-bit architecture Linux systems.

Further, the prediction model to be utilized can be specified: Net700 should be selected for Sanger reads, Net300 for reads shorter than 300 bp. For calculating the final combination of predicted genes per fragment, Orphelia by default allows a maximal overlap of 60 bp between genes. The maximal overlap can be varied through the

Table 1. Mean and standard deviation of sensitivity, specificity and harmonic mean on 300 and 700 bp DNA fragments that were randomly excised from 12 test species

	300 bp fragments			700 bp fragments		
	Sensitivity	Specificity	Harmonic mean	Sensitivity	Specificity	Harmonic mean
Orphelia Net300	82.1 ± 3.6	91.7 ± 3.8	86.6 ± 2.7	49.5 ± 13.8	79.3 ± 6.9	59.4 ± 10.2
Orphelia Net700	83.8 ± 3.4	88.1 ± 4.9	85.8 ± 3.9	88.4 ± 3.1	92.9 ± 3.2	90.6 ± 2.9
MetaGene	89.3 ± 3.3	84.2 ± 6.0	86.6 ± 4.3	92.6 ± 3.1	88.6 ± 5.9	90.4 ± 4.0
MetaGeneAnnotator	90.1 ± 2.8	86.2 ± 5.7	89.1 ± 3.1	92.9 ± 3.0	90.0 ± 6.0	91.5 ± 3.3
GeneMark	87.4 ± 2.8	91.0 ± 4.2	89.1 ± 3.1	90.9 ± 2.7	92.2 ± 5.1	91.5 ± 3.1

Orphelia Net300 represents Orphelia with the 300 bp prediction model, Orphelia Net700 represents the 700 bp prediction model. In addition, the performance of MetaGene, MetaGeneAnnotator and GeneMark is shown.

web interface. Finally, the user must provide a valid e-mail address to which an URL with a link to results will be sent.

Output

A typical run of Orphelia takes several minutes (for 10 MB input). Upon completion of the job, Orphelia sends an e-mail with two files to the user: the original input sequences, `seq.fna`, and the predicted genes, `gene.pred`. Predicted genes are given in a one-line-per-gene format:

```
>FragNo, GeneNo, Coord1_Coord2_Str_Fr_C_FH
```

`FragNo` is the fragment number in the input file, `GeneNo` is a numerical identifier of a Gene within the fragment. `Coord1` and `Coord2` indicate the positions of a predicted gene in the fragment, starting with position 1 at the beginning of the fragment. `Str` is the strand on which a predicted gene is encoded. The input sequence from 5′ to 3′ direction is assigned the '+' strand. `Fr` gives the reading frame of a gene counted from the 5′-end of the sequence. Reading frame 1 begins at the first nucleotide position of the input sequence, frame 2 at the second position and frame 3 at the third position. `C` is a label which indicates whether a candidate is complete (`C`) or incomplete (`I`). `FH` stands for the FASTA header of the input sequence. The first three entries are separated by a comma (`,`), all subsequent entries are separated by an underscore (_).

EXPERIMENTAL RESULTS

Evaluation on simulated data

We evaluated the accuracy of Orphelia's prediction models on DNA fragments of 300 and 700 bp, respectively. The fragments were randomly excised to a 10-fold genome coverage from 12 annotated test genomes that were not contained in the training set of Orphelia and that were first proposed by Noguchi *et al*. in 2006 (6). We measured sensitivity, which reflects how many of the existing genes were detected, and specificity, which shows how many of the predicted genes are annotated. In addition, the harmonic mean, which combines sensitivity and specificity within a single measure was used according to: $2 \times (\text{Sensitivity} \times \text{Specificity})/(\text{Sensitivity} + \text{Specificity})$. All predicted genes that match at least 60 bp in the same reading frame on the same strand with the annotation were counted as true positives. Table 1 shows the mean and standard deviation of performance over all species. Orphelia (Net700) has a prediction sensitivity of 88%, a specificity of 93% and a harmonic mean of 90.5% on 700 bp fragments. On 300 bp fragments (Net300), sensitivity (82%), specificity (92%) and the harmonic mean (86.6%) are lower than on 700 bp fragments, but the specificity is still very good.

In comparison to MetaGene, Orphelia has a lower sensitivity but shows a higher specificity, while the harmonic mean of both methods differs by <1%. The MetaGeneAnnotator shows a slightly higher harmonic mean than Orphelia and MetaGene, particularly on 300 bp fragments. Orphelia still has a higher specificity than the MetaGeneAnnotator. A direct comparison of GeneMark and Orphelia on the test setup shown here seems unfair if one considers that the model used by GeneMark was built using some of the test species. Keeping this in mind, GeneMark has a harmonic mean that is similar to MetaGeneAnnotator but has a specificity that is comparable with Orphelia.

In order to determine input sequence length-specific optimal models, we evaluated gene prediction accuracy of both models on fragments ranging in length from 200 bp to 500 bp in 20 bp intervals. The fragments were randomly excised to a 1-fold genome coverage from the test species mentioned above. While Net700 shows a softly decreasing sensitivity and specificity on shorter fragments, and a good performance on fragments as long as 60 000 bp (previously demonstrated in supplementary materials, Figure 4 of (9)), Net300 drastically drops in accuracy for fragments >300 bp. We therefore recommend the usage of Net300 for fragments ranging from 200 bp to 300 bp length, and Net700 for all longer fragments. More details can be seen in Supplementary Data, Figures 1 and 2.

In order to determine the effect of sequencing errors on gene prediction accuracy, several scenarios were simulated using the MetaSim software (10) and the same test species as before. We simulated error-free Sanger reads (with a mean length of 700 bp), and Sanger reads with error rates of 1×10^{-2}, 1×10^{-3}, 1×10^{-4} and 1×10^{-5} at the beginning of the read and error rates of 2×10^{-2}, 2×10^{-3}, 2×10^{-4} and 2×10^{-5} at the end of the read,

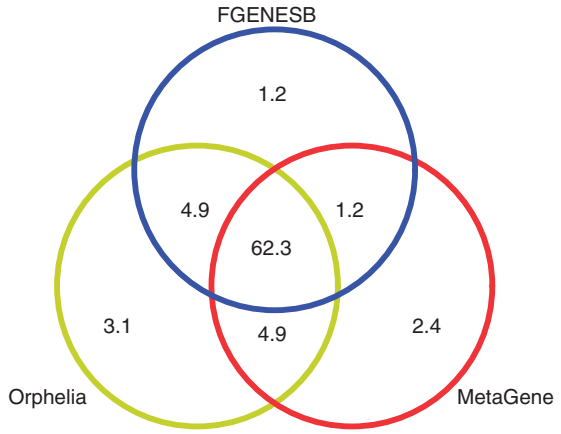


Figure 3. Venn diagram of the number of million nucleotides predicted as protein encoding by FGENESB, Orphelia (Net700) and MetaGene in the hypersaline microbial mat metagenome samples.

respectively (more details are given in the Methods section of Supplementary Data).

For a comprehensive evaluation of the effect of sequencing errors on gene prediction performance, predicted nucleotide sequences were translated to amino acid sequences using the standard translation table for prokaryotes. Predicted sequences were then aligned to annotated protein sequences using BLAT (11) with standard parameters. Matching amino acids were counted as true positives, amino acids that occur only in the annotation were counted as false negatives and amino acids that occur only in the prediction were counted as false positives. Based on these counts, we observe a decrease of sensitivity and specificity for Orphelia Net700 on Sanger reads with increasing error rates (see Supplementary Data, Table 1). For an error rate of $\sim 10^{-4}$, which was suggested by (6) as a realistic error rate, Orphelia shows a drop in accuracy of $<1\%$.

Application to real data

The hypersaline microbial mat metagenome consists of samples from 10 spatially successive layers of Guerrero Negro (12). Each sample was Sanger sequenced and contains ~13 000 reads. The original gene annotation of those reads was created with the commercial program FGENESB (http://www.softberry.com). Note that FGENESB integrates model-based gene prediction with homology-based annotation. In contrast to Orphelia and MetaGene, FGENESB also annotates rRNA and tRNA genes. For the following comparison of gene predictions, all RNA genes were removed from the FGENESB annotation.

We applied Orphelia (Net700) and MetaGene to the hypersaline microbial mat metagenome (all samples). The number of nucleotides that were predicted as protein encoding was counted and all possible intersections of nucleotides that were predicted as protein coding by Orphelia MetaGene, and FGENESB were calculated. The results are shown in Figure 3. All three methods predict $\sim 62.3 \times 10^6$ nt as protein coding. FGENESB predicts $\sim 1.2 \times 10^6$ nt, MetaGene predicts $\sim 2.4 \times 10^6$ nt and Orphelia predicts $\sim 3.1 \times 10^6$ nt as protein coding that were not predicted by any other method. FGENESB has an intersection of $\sim 4.9 \times 10^6$ nt with Orphelia, and an intersection of $\sim 1.2 \times 10^6$ nt with MetaGene. Both Orphelia and MetaGene predict about $\sim 4.9 \times 10^6$ nt as protein coding that were not predicted by FGENESB. Mavromatis *et al.* (13) reported FGENESB to overlook ~20% of the genes on single sequencing reads from annotated genomes, and that FGENESB 'newly predicted' ~10% genes in the same reads. We think that the intersection of nucleotides that were predicted by all methods contains highly reliable genes, and that at least the nucleotides commonly predicted by Orphelia and MetaGene, but not by FGENESB, are worth further investigation because they are likely to contain genes that were overlooked by FGENESB. Gene predictions of Orphelia and MetaGene on this dataset are available through the Orphelia web site.

IMPLEMENTATION

Orphelia's ORF finder is implemented in Java, while the ORF scoring routine and the greedy strategy for calculating the final gene combination are implemented in MATLAB using fast C ('mex') code for time critical subroutines. MATLAB routines are integrated as a MATLAB compiler generated program. The web server is based on Java Servlet technology. Submitted jobs are scheduled via a batch queuing system which allows simultaneous processing of several requests.

CONCLUSION

The evaluation on simulated data sets demonstrates that Orphelia shows high gene prediction accuracy on short DNA fragments and has—compared with the other web servers for metagenomic gene prediction—a particularly high gene prediction specificity. We showed that realistic sequencing error rates influence prediction performance only mildly. Therefore, the Orphelia web server application can be a valuable tool for predicting genes in metagenomic sequencing reads.

SUPPLEMENTARY DATA

Supplementary Data are available at NAR Online.

ACKNOWLEDGEMENTS

We thank Rasmus Steinkamp for helping us with the web server at GOBICS. We thank Dr Mario Stanke for discussions about the evaluation of gene predictions on the hypersaline microbial mat data set.

FUNDING

Georg-Christoph-Lichtenberg stipend granted by the state of Lower Saxony (to K.J.H.); fellowship within the Postdoc-program of the German Academic Exchange Service (DAAD to T.L.). Funding for open access

charge: Department for Bioinformatics, Institute for Microbiology and Genetics, Georg-August-Universität Göttingen.

Conflict of interest statement. None declared.

REFERENCES

1. Sanger,F., Nicklen,S. and Coulson,A.R. (1977) DNA sequencing with chain-terminating inhibitors. *Proc. Natl Acad. Sci. USA*, **74**, 5463–5467.
2. Ronaghi,M., Uhlén,M. and Nyreén,P. (1998) A sequencing method based on real-time pyrophosphate. *Science*, **281**, 363–365.
3. Altschul,S.F., Gish,W., Miller,W., Myers,E.W. and Lipman,D.J. (1990) Basic local alignment search tool. *J. Mol. Biol.*, **215**, 403–410.
4. Krause,L., Diaz,N.N., Bartels,D., Edwards,R.A., Pühler,A., Rohwer,F., Meyer,F. and Stoye,J. (2006) Finding novel genes in bacterial communities isolated from the environment. *Bioinformatics*, **22**, e281–e289.
5. Yooseph,S., Li,W. and Sutton,G. (2008) Gene identification and protein classification in microbial metagenomic sequence data via incremental clustering. *BMC Bioinformatics*, **9**, 182.
6. Noguchi,H., Park,J. and Takagi,T. (2006) MetaGene: prokaryotic gene finding from environmental shotgun sequences. *Nucleic Acids Res.*, **34**, 5623–5630.
7. Noguchi,H., Taniguchi,T. and Itoh,T. (2008) MetaGeneAnnotator: detecting species-specific patterns of ribosomal binding site for precise gene prediction in anonymous prokaryotic and phage genomes. *DNA Res.*, **15**, 387–396.
8. Besemer,J. and Borodovsky,M. (1999) Heuristic approach to deriving models for gene finding. *Nucleic Acids Res.*, **27**, 3911–3920.
9. Hoff,K.J., Tech,M., Lingner,T., Daniel,R., Morgenstern,M. and Meinicke,P. (2008) Gene prediction in metagenomic fragments: a large scale machine learning approach. *BMC Bioinformatics*, **9**, 217.
10. Richter,D.C., Ott,F., Auch,A.F., Schmid,R. and Huson,D.H. (2008) MetaSim – a sequencing simulator for genomics and metagenomics. *PLoS ONE*, **3**, e3373.
11. Kent,W.J. (2002) BLAT – the BLAST-like alignment tool. *Genome Res.*, **12**, 656–664.
12. Kunin,V., Raes,J., Harris,J.K., Spear,J.R., Walker,J.J., Ivanova,N., von Mering,C., Bebout,B.M., Pace,N.R., Bork,P. *et al.* (2008) Millimeter-scale genetic gradients and community level molecular convergence in a hypersaline microbial mat. *Mol. Syst. Biol.*, **4**, 198.
13. Mavromatis,K., Ivanova,N., Barry,K., Shapiro,H., Goltsman,E., McHardy,A.C., Rigoutsos,I., Salamov,A., Korzeniewski,F., Land,M. *et al.* (2007) Use of simulated data sets to evaluate the fidelity of metagenomic processing methods. *Nat. Methods*, **4**, 1548–7091.

Chapter 5

The Effect of Sequencing Errors on Metagenomic Gene Prediction

Citation

K. J. Hoff.
The effect of sequencing errors on metagenomic gene prediction.
BMC Genomics 2009 10:520
doi:10.1186/1471-2105-9-217

BMC Genomics

BioMed Central

The effect of sequencing errors on metagenomic gene prediction

BMC Genomics 2009, **10**:520 doi:10.1186/1471-2164-10-520

Katharina J Hoff (katharina@gobics.de)

ISSN 1471-2164

Article type Research article

Submission date 25 June 2009

Acceptance date 12 November 2009

Publication date 12 November 2009

Article URL http://www.biomedcentral.com/1471-2164/10/520

The effect of sequencing errors on metagenomic gene prediction

Katharina J Hoff[1,2*]

[1]Department of Bioinformatics, Institute of Microbiology and Genetics, Georg-August-University Göttingen, Göttingen, Germany
[2]International Max Planck Research School for Molecular Biology, Georg-August-University Göttingen, Göttingen, Germany

Email: Katharina J Hoff*- katharina@gobics.de;

*Corresponding author

Abstract

Background: Gene prediction is an essential step in the annotation of metagenomic sequencing reads. Since most metagenomic reads cannot be assembled into long contigs, specialized statistical gene prediction tools have been developed for short and anonymous DNA fragments, e.g. MetaGeneAnnotator and Orphelia. While conventional gene prediction methods have been subject to a benchmark study on real sequencing reads with typical errors, such a comparison has not been conducted for specialized tools, yet. Their gene prediction accuracy was mostly measured on error free DNA fragments.

Results: In this study, Sanger and pyrosequencing reads were simulated on the basis of models that take all types of sequencing errors into account. All metagenomic gene prediction tools showed decreasing accuracy with increasing sequencing error rates. Performance results on an established metagenomic benchmark dataset are also reported. In addition, we demonstrate that ESTScan, a tool for sequencing error compensation in eukaryotic expressed sequence tags, outperforms some metagenomic gene prediction tools on reads with high error rates although it was not designed for the task at hand.

Conclusions: This study fills an important gap in metagenomic gene prediction research. Specialized methods are evaluated and compared with respect to sequencing error robustness. Results indicate that the integration of error-compensating methods into metagenomic gene prediction tools would be beneficial to improve metagenome annotation quality.

Background

Metagenomes are analyzed through simultaneous sequencing of all species in a microbial community without prior cultivation under laboratory conditions. The result is usually a large collection of sequencing reads from many species, and the phylogenetic origin of each read is unknown. A major goal in all metagenomic studies is the identification of potential protein functions and metabolic pathways. Reliable gene predictions are the basis for correct functional annotation, and for the discovery of new genes with their functions.

Several gene prediction methods have been developed for the *ab initio* identification of protein coding

genes in complete microbial genomes (e.g. GLIMMER and GeneMark [1, 2]). These methods require an initial training phase on some data from the target genome, or training on the genome of a closely related species. Such *conventional* gene finders can in principle be applied to metagenomic data, given that single sequencing reads can be *assembled* into longer contigs in order to provide sufficient training data. The applicability of *conventional* gene finders to metagenomic contigs can be improved by *binning* contigs and reads into separate phylogenetic scaffolds, e.g. by their oligonucleotide signature [3]. However, the assembly of metagenomic sequencing reads is problematic. Mavromatis *et al.* (2007) demonstrated on artificial metagenomes that assembly quality highly depends on the sequencing coverage of single species within the metagenome [4]. They also showed that short contigs are at high risk of chimerism, i.e. a read from species A is joined with a read of species B, which limits the use of contigs for further analysis. Some proportion of most metagenomes remains in single unassigned sequencing reads after assembly and binning, and in some cases, metagenome assembly fails completely, e.g. for the hypersaline microbial mat metagenome [5]. For this reason, the ability of predicting genes in single and anonymous sequencing reads is essential to fully explore a metagenome.

This problem can be solved by two strategies. One possibility is the identification of protein coding regions through sequence similarity. An example is to conduct a BLAST search [6] with metagenomic sequences against a database of known proteins. Annotation success is here limited to already known genes and their close relatives. This problem is particularly prominent for viral sequences that are poorly represented in databases [7–9]. Clustering of open reading frames (ORFs) in principle enables sequence similarity based methods to identify novel genes that are conserved within the metagenomic sample [10, 11]. Considering the size of most metagenomes, computational cost is a limiting factor for these methods.

A different strategy is based on gene prediction with statistical models. GeneMark with heuristic models [12], MetaGene [13], Orphelia [14, 15] and MetaGeneAnnotator [16] fall into the category of model-based metagenomic gene prediction tools. The common advantage of these tools is the capability to predict known and novel genes at a lower computational cost. Their mostly unexplored disadvantage is the susceptibility to sequencing errors - which methods that are based on sequence similarity may automatically compensate to a certain extent.

The possible effect of sequencing errors on model-based metagenomic gene prediction depends on the actual error rates. The two major sequencing techniques that are commonly used in metagenomics have different sequencing accuracy. Chain termination sequencing [17] was the first method to be used for metagenome sequencing. It produces an average read length of $\sim$700 nucleotides (nt). The error rates

reported for Sanger sequencing vary from 0.001% [13, 18] to more than 1% [19, 20] and seem to depend on the software that is used for post processing of reads. Pyrosequencing, also known as "454 sequencing", produces shorter reads [21, 22]. In the beginning, read length was about 110 nt and has now increased to more than ∼450 nt. Huse *et al.* (2007) reported an error rate of 0.49% for reads of the length 100-200 nt [23], the read simulation software MetaSim [20] produces reads with an error rate of 2.8% with parameters that are adjusted according to an original 454 publication [22]. Pyrosequencing is still subject to constant research. In the near future, a further increase in read length can be expected.

For all techniques, sequencing accuracy is high at the beginning of a read and decreases with read length. Three error types can occur: (1) substitution errors, that means a wrong nucleotide is read out, (2) deletion errors, in which one or more nucleotides are omitted, and (3) insertion errors, where one or more nucleotides are falsely added to the sequence during the reading process.

All statistical gene prediction tools utilize codon usage as an important feature to identify protein coding genes. If a nucleotide is deleted or inserted into the sequence, this causes a shift in the reading frame. Methods that do not compensate for frame shifts cannot predict affected genes accurately. Substitution errors will only affect one codon and their influence on gene prediction accuracy is therefore generally smaller. All types of errors may also result in additional stop codons. False stop codons may have an even more severe effect on gene prediction than a frame shift because they will definitely terminate a predicted gene.

The robustness with respect to sequencing errors in Sanger reads has been investigated and discussed for MetaGene and Orphelia [13, 14], other tools have not been evaluated with regard to this property. In particular, no studies about the effect of sequencing errors in 454 reads on metagenomic gene prediction are available. Three benchmark data sets that were supposed to facilitate the accuracy evaluation of metagenome analysis tools on real data were introduced [4] but so far, metagenomic gene prediction tools have not been evaluated on these data sets.

In this work, we demonstrate the extent to which typical errors in Sanger and pyrosequencing reads affect metagenomic gene prediction. The effect strongly depends on the actual error rate. For investigation, we utilize sequences simulated with MetaSim, a metagenome simulator [20]. Gene prediction quality on the metagenomic benchmark data sets is also shown and discussed. ESTScan [24], a tool for the curation of expressed sequence tags (ESTs), was trained for the application to metagenomes, and gene prediction accuracy results of ESTscan lead us to the conclusion that the integration of error compensating methods into metagenomic gene prediction tools might significantly improve their performance, and with this

metagenome annotation quality.

Results

Simulated reads

The evaluation of metagenomic gene prediction tools is complicated by a lack of reliably annotated metagenomic reads. The annotation quality of complete genomes can be expected to be much better. For this reason, we used simulated reads from annotated genomes for the evaluation of metagenomic gene prediction tools.

The models underlying all metagenomic gene prediction tools were built on the basis of genomes from selected training species. Generalization capabilities of those models can only be analyzed if training species and their close relatives (we define *close relative* as *species from the same genus*) are excluded from the evaluation setup. In this study, we did not aim at the simulation of realistic microbial communities, but instead, we wanted to encompass a wide range of phyla. Therefore, a set of prokaryotic microorganisms was selected according to this criterion (see Table 1). None of these species has a genus relative in the training data of metagenomic gene predictions tools investigated in this study.

Sanger sequencing reads with the average length of 700 nt were simulated with error rates ranging from 0 to 1.5%, and 454 reads that are on average 450 nt long were simulated with error rates ranging from 0 to 2.8%. All simulated reads are available at http://metagenomic-benchmark.gobics.de.

ESTScan matrix

ESTScan is a tool that was originally developed to detect coding regions in eukaryotic ESTs and simultaneously correct sequencing errors [24]. In contrast to metagenomic gene prediction methods, ESTScan cannot detect overlapping coding regions as they frequently occur in prokaryotic genomes. We were interested in ESTScan's sequencing error correction capabilities. In order to test if ESTScan would also compensate errors in coding regions on metagenomic sequencing reads, we trained an ESTScan scoring matrix on prokaryotic genomes that were also used for training MetaGene and Orphelia. The matrix is availabe at http://metagenomic-benchmark.gobics.de.

Accuracy on unassembled simulated reads

Gene prediction accuracy can be estimated by measuring the overlap of predicted and annotated genes in the same reading frame. A comparison on amino acid sequence level reflects overlap and reading frame if

the basic requirement that the sequences are highly similar (almost identical) is fulfilled. An amino acid sequence alignment with only few missmatches and gaps shows this kind of similarity. We used BLAT [25] alignments of predicted and annotated genes to assess gene prediction accuracy. Three classes of genes were defined: (1) predicted genes that had a BLAT alignment of at least 20 amino acids (aa) length and at least 80% sequence identity were called true positives, (2) annotated genes that did not fall into the first category were counted as false negatives, and (3) predicted genes that did not have a match with the annotation according to the first criterion were counted as false positives. With these counts, we measured the proportion of annotated genes that were predicted (sensitivity) and the proportion of predicted genes that match genes in the annotation (specificity). Further details are given in section Methods.

On simulated Sanger sequencing reads, MetaGene and MetaGeneAnnotator show the highest gene prediction sensitivities ($\sim$ 94% over all species on error free reads and $\sim$80% on reads with the highest error rate) while Orphelia has the best specificity values with $\sim$96% on error free reads and $\sim$92% on reads with the highest error rate (compare Table 2). This result is in agreement with previous publications [14, 15].

To estimate overall gene prediction accuracy, sensitivity and specificity were combined in a harmonic mean. Results are visualized in Figure 1. Generally, the MetaGeneAnnotator shows the highest accuracy. The accuracy of all tools decreases only very mildly (by $\sim$1.4%) on reads from 0.0015% to 0.15% errors. Also common to all tools is a drastic drop in accuracy of $\sim$10% from reads with 0.15% errors to reads with 1.5%.

For a concise picture, we also measured amino acid prediction accuracy. For this, all amino acids that were captured into a BLAT alignment of prediction and annotation were counted as true positives. All amino acids in the annotation that were not predicted by the true positive criterion were counted as false negatives, and the remaining predicted amino acids were counted as false positives.

All tools have an amino acid sensitivity of $\sim$95 to 96% on error free reads. Orphelia and MetaGeneAnnotator have with 97% the highest specificity [see Additional file 1, Table S1]. These values are higher than the corresponding gene prediction accuracy, indicating that long genes on error free reads are more likely to be predicted correctly than short genes. On reads with high error rates (0.15% and 1.5%), we observed that gene prediction rates were higher than amino acid prediction rates. The reason is that long open reading frames are likely to be affected by sequencing errors in form of in frame stop codons. Consequently, only shorter genes can be predicted, which is also confirmed by length boxplots of genes predicted in reads with different error rates [see Additional file 1, Figure S1].

ESTScan had with $\sim$87% a lower harmonic mean (gene prediction level) than metagenomic gene prediction tools on reads with few errors (see Figure 1). Interestingly, the decrease in performance on reads with

0.015% errors to 0.15% errors was only 0.7%, which is smaller than the decrease observed for metagenomic gene prediction tools (ranging from 0.8 to 1.4%). From reads with 0.15% to 1.5% errors, a decrease in accuracy of 9% was observed, which is also a smaller accuracy drop than measured for metagenomic gene prediction tools.

On error free 450 nt pyrosequencing reads, gene prediction sensitivity and specificity of all tools is similar to accuracy on Sanger reads (see Table 3). Also here, MetaGeneAnnotator has the highest gene prediction harmonic mean (94%) as depicted in Figure 2. From error free reads to reads with 0.49% errors, a drop in harmonic mean of ∼9% is observed for all gene prediction methods (except for Orphelia with 12%). Opposed to this, ESTScan again showed a smaller decrease of 7%. Continuing to reads with an error rate of 2.8%, a further accuracy decrease of ∼35 (MetaGeneAnnotator, MetaGene, GeneMark) to ∼42% (Orphelia and ESTScan) follows. On amino acid level, we observe the same effects as for Sanger reads [see Additional file 1, Table S2].

On the example of 454 reads with an error rate of 0.49%, we further investigated to which extent the GC-content of a read influences gene prediction accuracy. (GC-content is the percentage of bases cytosine and guanine in all bases of a sequence.) Table 4 shows that GeneMark has a higher gene prediction accuracy on low-GC species than on species with a high GC-content. For MetaGene and MetaGeneAnnotator, this effect is smaller, and for Orphelia and ESTScan, we can not find an obvious difference. On the same dataset, we also measured gene prediction sensitivity and specificity for different gene lengths. For measuring sensitivity, we evaluated predicted genes with all lengths against annotated genes of certain length categories (up to 40 aa length, 41 to 80, 81 to 120, 121 to 160, 161 to 200). Specificity was measured by evaluating predicted genes of the length categories against annotated genes of all lengths. All tools most accurately predict genes of 160 aa length or longer (those genes mostly span the complete read). MetaGene and MetaGeneAnnotator have the highest sensitivity on shorter genes while Orphelia has the highest specificity among gene prediction tools on shorter genes [see Additional file 1, Figure S2].

Accuracy on FAMeS reads

The "Fidelity of Analysis of Metagenomic Samples" (FAMeS) benchmark data sets consist of sequencing reads from single species genome projects [4]. The benchmark data sets were designed to measure the accuracy of assemblers, binning methods and gene prediction methods. Particularly for assessing assembly and binning accuracy, the reads were combined into three sets with different degrees of representation for

each species. The low-complexity data set (simLC) consists of reads from mostly one species with a few reads from less abundant species. The medium-complexity data set (simMC) resembles a moderately complex community with more than one dominant population and also has few reads from less abundant species. The high-complexity data set (simHC) lacks dominant populations. However, detecting a difference in gene prediction accuracy between the three different sets will only show that a tool is better for predicting genes in one species than in others. We used the unassembled reads of all FAMeS data sets to test gene prediction accuracy of tools that were mainly designed for the application to single reads or short contigs. Regarding the results, one must consider that the sequence quality of FAMeS raw reads is rather low because the reads are untrimmed, meaning that their low-quality ends have not been removed. On all three data sets, GeneMark shows the highest harmonic mean (∼81%) [see Additional file 1, Figure S3]. The most sensitive method on all data sets was MetaGene (78% on simMC and simHC, 80% on simLC), and the highest specificities were observed for GeneMark (86% on simLC, 85% on simMC, 83% on simHC, see Table 5).

Interestingly, ESTScan performs almost as good as GeneMark on FAMeS reads (most likely due to the low sequence quality of the FAMeS data set).

Discussion

The major question of this study was how sequencing errors affect metagenomic gene prediction accuracy. On Sanger reads, prediction accuracy is only mildly affected by error rates of up to 0.15%. Sequencing accuracy in this range is realistic for most combinations of chain termination sequencers and read postprocessing software. In general, all tools are therefore applicable to metagenomic Sanger sequences. The FAMeS data set gives an example for low quality Sanger sequences. On those reads, the accuracy of specialized metagenomic gene prediction tools is not much higher than the performance of conventional gene prediction tools (see [4]). The quality of these reads is very likely to be improved by read postprocessing steps, e.g. by trimming read ends. The results on the FAMeS data set demonstrate that such postprocessing steps are important to ensure a high gene prediction accuracy.

In 2008, Wommack *et al.* showed that a read length of 400 nt significantly weakens BLAST analysis results from metagenomic data [9]. For gene prediction tools that use statistical models, our findings on simulated 450 nt pyrosequencing reads suggest that the actual read length of 450 nt has almost no impact on gene prediction accuracy results when compared to accuracy on 700 nt Sanger reads (compare Tables 2 and 3). Sequencing errors lead to a decrease in gene prediction accuracy, though. On reads with a realistic error

rate of 0.49% for pyrosequencing, the harmonic mean is around 82%, which leaves much room for improvement. The identification of short gene fragments, particularly gene fragments shorter than 120 aa (360 nt), is a major shortcoming of metagenomic gene prediction tools. Therefore, gene prediction in pyrosequencing reads would benefit from further development of models specialized on the accurate detection of short gene fragments.

Another interesting question is whether metagenomic gene prediction tools that were trained on a limited set of species genomes are able to predict genes in reads from distantly related species, and whether it is possible to name a 'best tool' for this purpose. The simulated reads used in this study were sampled from species that belong to many different phyla. Members of some phyla were used for training of all tools, other phyla were not represented by a training species. We show that gene prediction accuracy varies over reads from different test species but we believe that this variation is independent from the degree of relatedness to training species. *Dictyoglomus thermophilum* is an example whose phylum was excluded from training of all tools. No significant drop in accuracy can be observed for reads from this species (see Table 4).

On the simulated reads here, it looks like MetaGeneAnnotator is the best tool. In contrast to this, GeneMark has the highest accuracy on FAMeS reads - which are constituted from different species than the simulated data set. Also the results of Hoff *et al.* (2009) demonstrated a high prediction accuracy for GeneMark on a different simulated data set [15]. From the presented data, it is not possible to conclude whether metagenomic gene prediction tools work better or worse for reads from particular phyla because most phyla are represented by only one species. However, it seems that gene prediction accuracy of single tools depends on the species contained in a metagenome.

The accuracy of metagenomic gene prediction tools on real sequencing reads affects further steps of metagenome analysis that generally depend on the predictions. For example, the functional annotation of genes is often achieved by using HMMER [26] or fast tools like UFO [27] for sensitive protein domain database search, e.g. to detect Pfam domains (e.g. [28,29]). The profile hidden Markov models for HMMER or the UFO algorithm cannot detect domains correctly if the predicted genes that are used for database search are affected by frame shift errors within the domain region. In addition, gene prediction sensitivity directly influences the number domains that can potentially be detected. A high gene prediction accuracy ensures that such gene prediction dependent steps of analysis can also be carried out with high accuracy. We showed that ESTScan, although not designed for metagenomic gene prediction, is in principle capable of compensating for frame shift errors in metagenomic data to some extent. Therefore, metagenomic gene

prediction accuracy could be greatly improved by the integration of methods that are robust with respect to sequencing errors.

Conclusions

In conclusion, the integration of error compensating methods into metagenomic gene prediction tools as well as the development of suitable models specialized on the accurate detection of short gene fragments would be beneficial to improve metagenome annotation quality.

Methods

Read simulation

Sanger and 454 sequencing reads were simulating with MetaSim from the genomes of species given in Table 1 with 1-fold genome coverage as defined in [14, 15].

Sanger reads were simulated with the error rates 0%, 0.0001% at the read start and 0.0002% at the read end, 0.001% to 0.002%, 0.01% to 0.02%, 0.1% to 0.2%, and 1% to 2%, and an average read length of 700 nt.

Pyrosequencing reads were simulated with MetaSim from the same genomes, with an average read length of 450 nt, and with the error rates 0%, 0.22%, 0.49%, and 2.8% (1-fold genome coverage).

Further details on the simulation parameters of MetaSim are given in Additional file 1, section Supplementary methods.

Benchmark data set

The FAMeS benchmark data sets simLC, simMC and simHC were retrieved from http://fames.jgi-psf.org/Retrieve_data.html in September 2008. For gene prediction accuracy assessment, the "genes that are included in the reference genomes" (further referred to as amino acid annotation file), and the "overlap of the genes with the sequencing reads" were also downloaded. The amino acid annotation file contains the full length amino acid sequences from genes in all genomes.

Gene prediction

Genes in simulated reads and reads of the FAMeS data set were predicted with Genemark heuristic version 1.1, MetaGene and MetaGeneAnnotator as provided at http://metagene.cb.k.u-tokyo.ac.jp/metagene on February 1st 2009, respectively, and Orphelia as provided at http://orphelia.gobics.de/download.jsp on May 1st 2009. Concerning the two different models of Orphelia, we applied Net300 to all reads shorter or equal the length of 300 nt. Net700 was used for all remaining reads.

Genemark was run with the parameter `-a` to produce an amino acid sequence output. Amino acid sequences for the other tools were translated from nucleotide sequences that were excised from the simulated reads according to the predicted gene coordinates using BioPerl with the standard translation table [30].

ESTScan training

For the application to metagenomic data, ESTScan 2.1 (available at http://sourceforge.net/projects/estscan) was trained with the annotated genes from the training genomes of MetaGene and Orphelia. Full coding regions were excised from the genomes with a flanking region of 50 nt. To directly obtain predicted amino acid sequences, ESTScan was applied to simulated and benchmark data with the option `-t`.

Accuracy Assessment

Gene prediction accuracy was assessed through the alignment of amino acid sequences with BLAT. For simulated reads, the translation of annotated protein coding genes in the error free version of simulated reads were used as a reference. For the benchmark data set, full length amino acid sequences that completely or partically overlap with the reads were used as a reference.
True positives, false negatives and false positives are described in section 'Results', paragraph 'Accuracy on simulated genes'. Sensitivity and specificity are defined in equations 1 and 2:

$$\text{sensitivity} = \frac{\text{true positives}}{(\text{true positives} + \text{false negatives})} \tag{1}$$

$$\text{specificity} = \frac{\text{true positives}}{(\text{true positives} + \text{false positives})} \tag{2}$$

Sensitivity and specificity were combined into one measure by the harmonic mean (3):

$$\text{harmonic mean} = \frac{2 \times \text{sensitivity} \times \text{specificity}}{\text{sensitivity} + \text{specificity}} \tag{3}$$

Authors' contributions

KJH designed, conducted and evaluated all experiments. The author wrote, read and approved the final manuscript.

Appendix: footnotes

[1] Error rates are given as 'error rate at the read start' to 'error rate at the read end'.

[2] Sensitivity (Sens.) expresses how many of the annotated genes were predicted.

[3] Specificity (Spec.) shows how many of the predicted genes were true.

[4] Low-complexity simulated data set with one dominating species and few reads from less abundant species.

[5] Medium-complexity simulated data set with several dominating species and reads form less abundant species.

[6] High-complexity simulated data set without dominating species.

Acknowledgements

I would like to thank Dr. Peter Meinicke, Dr. Maike Tech, and Dr. Mario Stanke for fruitful discussions about this study, Fabian Schreiber for programming assistance, Dr. Rolf Daniel for advice concerning current sequencing error rates, and Dr. Peter Meinicke for proofreading of the manuscript. I am also very grateful for the instructive comments of two anonymous reviewers.

KJH was financially supported by a Georg-Christoph-Lichtenberg stipend granted by the State of Lower Saxony, Germany.

References

1. Delcher AL, Harmon D, Kasif S, White O, Salzberg SL: **Improved microbial gene identification with GLIMMER**. *Nucleic Acids Res.* 1999, **27**(23):4636–4641.
2. Lukashin A, Borodovsky M: **GeneMark.hmm: new solutions for gene finding**. *Nucleic Acids Res.* 1998, **26**(4):1107–1115.
3. Woyke T, Teeling H, Ivanova N, Huntemann M, Richter M, Gloeckner F, Boffelli D, Anderson I, Barry K, Shapiro H, Szeto E, Kyrpides N, Mussmann M, Amann R, Bergin C, Ruehland C, Rubin E, Dubilier N: **Symbiosis insights through metagenomic analysis of a microbial consortium**. *Nature* 2006, **443**:950–955.
4. Mavromatis K, Ivanova N, Barry K, Shapiro H, Goltsman E, McHardy A, Rigoutsos I, Salamov A, Korzeniewski F, Land M, Lapidus A, Grigoriev I, Richardson P, Hugenholtz P, Kyrpides N: **Use of simulated data sets to evaluate the fidelity of metagenomic processing methods**. *Nat. Meth.* 2007, **4**(6):1548–7091.
5. Kunin V, Raes J, Harris J, Spear J, Walker J, Ivanova N, von Mering C, Bebout B, Pace N, Bork P, Hugenholtz P: **Millimeter-scale genetic gradients and community level molecular convergence in a hypersaline microbial mat**. *Molecular Systems Biol.* 2008, **4**:198.
6. Altschul S, Gish W, Miller W, Myers E, Lipman D: **Basic local alignment search tool**. *J. Mol. Biol.* 1990, **215**:403–410.
7. Angly F, Felts B, Breitbart M, Salamon P, Edwards R, Carlson C, Chan A, Haynes M, Kelley S, Liu H, Mahaffy J, Mueller J, Nulton J, Olson R, Parsons R, Rayhawk S, Suttle C, Rohwer F: **The Marine Viromes of Four Oceanic Regions**. *PLoS Biol.* 2006, **4**(11):e368.
8. Bench S, Hanson T, Williamson K, Ghosh D, Radosovich M, Wang K, Wommack K: **Metagenomic Characterization of Chesapeake Bay Virioplankton**. *Appl Environ Microbiol.* 2007, **73**(23):7629–7641.
9. Wommack K, Bhavsar J, Ravel J: **Metagenomics: read length matters**. *Appl Environ Microbiol.* 2008, **74**(5):1453–1463.
10. Krause L, Diaz NN, Bartels D, Edwards RA, Pühler A, Rohwer F, Meyer F, Stoye J: **Finding novel genes in bacterial communities isolated from the environment**. *Bioinformatics* 2006, **22**(14):e281–e289.

11. Yooseph S, Sutton G, Rusch D, Halpern A, Williamson S, Remington K, Eisen J, Heidelberg K, Manning G, Li W, Jaroszewski L, Cieplak P, Miller C, Li H, Mashiyama S, Joachimiak M, van Belle C, Chandonia JM, Soergel DA, Zhai Y, Natarajan K, Lee S, Raphael B, Bafna V, Friedmann R, Brenner SE, Godzik A, Eisenberg D, Dixon J, Taylor S, gand M Frazier RS, Venter J: **The Sorcerer II global ocean sampling expedition: expanding the universe of protein families**. *PLoS Biology* 2007, **5**(3):0432–0466.

12. Besemer J, Borodovsky M: **Heuristic approach to deriving models for gene finding**. *Nucleic Acids Res.* 1999, **27**(19):3911–3920.

13. Noguchi H, Park J, Takagi T: **MetaGene: prokaryotic gene finding from environmental shotgun sequences**. *Nucleic Acids Res.* 2006, **34**(19):5623–5630.

14. Hoff KJ, Tech M, Lingner T, Daniel R, Morgenstern M, Meinicke P: **Gene prediction in metagenomic fragments: A large scale machine learning approach**. *BMC Bioinformatics* 2008, **9**:217.

15. Hoff KJ, Lingner T, Meinicke P, Tech M: **Orphelia: predicting genes in metagenomic sequencing reads**. *Nucleic Acids Res.* 2009, **37**:W101–W105.

16. Noguchi H, Taniguchi T, Itoh T: **MetaGeneAnnotator: detecting species-specific patterns of ribosomal binding site for precise gene prediction in anonymous prokaryotic and phage genomes**. *DNA Res* 2008, **15**(6):387–396, [http://dnaresearch.oxfordjournals.org/cgi/content/abstract/15/6/387].

17. Sanger F, Nicklen S, Coulson AR: **DNA sequencing with chain-terminating inhibitors**. *Proc. Natl. Acad. Sci. USA* 1977, **74**(12):5463–5467.

18. Ewing B, LaDeana H, Wendl MC, Green P: **Base-calling of automated sequencer traces using Phred I. accuracy assessment**. *Genome Res.* 1998, **8**:175–185.

19. Keith CS, Hoang DO, Barrett BM, Feigelman B, Nelson MC, Thai H, Baysdorfer C: **Partial sequence analysis of 130 randomly selected Maize cDNA clones**. *Plant Physiol.* 1993, **101**:329–332.

20. Richter DC, Ott F, Auch AF, Schmid R, Huson DH: **MetaSim − a sequencing simulator for genomics and metagenomics**. *PLoS ONE* 2008, **3**(10):e3373.

21. Ronaghi M, Uhlén M, Nyreén P: **A sequencing method based on real-time pyrophosphate**. *Science* 1998, **281**(5375):363–365.

22. Margulies M, Egholm M, Altman WE, Attiya S, Bader J, Bemben L, Berka J, Braverman M, Chen YJ, Chen Z, Dewell S, Du L, Fierro JM, Gomes X, Godwin BC, He W, Helgesen S, Ho C, Irzyk G, Jando S, Alenquer M, Jarvie T, Jirage K, Kim JB, Knight J, Lanza J, Leamon J, Lefkowitz S, Lei M, Li J, Lohman K, Lu H, Makhijani V, McDade K, McKenna M, Myers E, Nickerson E, Nobile J, Plant R, Puc B, Ronan M, Roth G, Sarkis G, Simons J, Simpson J, Srinivasan M, Tartaro K, Tomasz A, Vogt K, Volkmer G, Wang S, Wang Y, Weiner M, Yu P, Begley R, Rothberg J: **Genome sequencing in microfabricated high-density picolitre reactors**. *Nature* 2005, **437**:376–380.

23. Huse SM, Huber JA, Morrison HG, Sogin ML, Welch DM: **Accuracy and quality of massively parallel DNA pyrosequencing**. *Genome Biol.* 2007, **8**:R143.

24. Lottaz C, Iseli C, Jongeneel CV, Bucher P: **Modeling sequencing errors by combining Hidden Markov models**. *Bioinformatics* 2003, **19**:ii103–ii112.

25. Kent WJ: **BLAT − the BLAST-like alignment tool**. *Genome Res.* 2002, **12**(4):656–664.

26. Durbin R, Eddy SR, Mitchison G: *Biological sequence analysis: probabilistic models of proteins and nucleic acids*. Cambridge: Cambridge University Press 1998.

27. Meinicke P: **UFO: a web server for ultra-fast functional profiling of whole genome protein sequences**. *BMC Genomics* 2009, **10**:409.

28. Harrington E, Singh A, Doerks T, Letunic I, von Mering C, Jensen L, Raes J, Bork P: **Quantitative assessment of protein function prediction from metagenomics shotgun sequences**. *Proc. Natl. Acad. Sci. USA* 2007, **104**(35):13913–13918.

29. Gilbert J, Field D, Huang Y, Edwards R, Li W, Gilna P, Joint I: **Detection of large numbers of novel sequences in the metatranscriptomes of complex marine microbial communities**. *PLoS ONE* 2008, **3**(8):e3042.

30. Stajich J, Block D, Boulez K, Brenner S, Chervitz S, Dagdigian C, Fuellen G, Gilbert J, Korf I, Lapp H, Lehväslaiho H, Matsalla C, Mungall C, Osborne B, Pocock M, Schattner P, Senger M, Stein L, Stupka E, Wilkinson M, Birney E: **The Bioperl toolkit: Perl modules for the life sciences.** *Genome Res.* 2002, **12**(10):1611–1618.

Figures

Figure 1 - Average gene prediction accuracy on simulated Sanger reads.

The harmonic mean is a measure that combines sensitivity and specificity (mean and standard deviation over all species in the simulated metagenome are shown).

Figure 2 - Average gene prediction accuracy on simulated 454 reads.

The harmonic mean is a measure that combines sensitivity and specificity (mean and standard deviation over all species in the simulated metagenome are shown).

Tables

Table 1 - Test species.

Species whose genomes were used to simulate sequencing reads.

Species	Phylum	GC-content
Acholeplasma laidlawii PG-8A	Termicutes	31%
Buchnera aphidicola str. APS	Proteobacteria (β)	30%
Burkholderia pseudomallei K96243	Proteobacteria (γ)	68%
Chlorobium tepidum TLS	Bacteriodetes/Chlorobi group	56%
Corynebacterium jeikeium K411	Actinobacteria	61%
Desulfurococcus kamchatkensis 1221n	Crenarcheota	45%
Dictyoglomus thermophilum H-6-12	Dictyoglomi	33%
Exiguobacterium sibiricum 255-15	Firmicutes	47%
Herpetosiphon aurantiacus ATCC 23779	Chloroflexi	50%
Hydrogenobaculum sp. Y04AAS1	Aquificae	34%
Natronomonas pharaonis DSM 2160	Euryarchaeota	63%
Nitrosopumilus maritimus SCM1	Crenarcheota	34%
Prochlorococcus marinus str. MIT 9312	Cyanobacteria	31%
Wolbachia endosymbiont strain TRS of Brugia malayi	Proteobacteria (α)	34%

Table 2 - Accuracy on simulated Sanger reads.
The gene prediction accuracy (mean and standard deviation over all species in the simulated metagenome) results of four metagenomic gene prediction tools GeneMark, MetaGene, MetaGeneAnnotator (MGA) and Orphelia, and of the EST processing tool ESTScan on simulated Sanger reads is shown.

	GeneMark		MetaGene		MGA		Orphelia		ESTScan	
Error rate[1]	Sens.[2]	Spec.[3]	Sens.[2]	Spec.[3]	Sens.[2]	Spec.[3]	Sens.[2]	Spec.[3]	Sens.[2]	Spec.[3]
0 to 0	91.9 ± 3.2	93.8 ± 4.9	94.4 ± 3.0	93.0 ± 2.9	94.7 ± 2.9	94.1 ± 2.9	89.7 ± 3.5	96.5 ± 1.7	78.9 ± 7.2	98.5 ± 1.2
1 to 2×10^{-5}	91.9 ± 3.3	93.7 ± 5.2	94.8 ± 2.8	93.0 ± 3.0	94.8 ± 2.9	94.0 ± 3.1	90.1 ± 3.3	96.7 ± 1.6	79.2 ± 6.5	98.6 ± 1.1
1 to 2×10^{-4}	91.8 ± 3.3	93.5 ± 5.2	94.5 ± 2.9	92.6 ± 3.2	94.5 ± 3.0	93.7 ± 3.1	89.6 ± 3.5	96.5 ± 1.7	79.0 ± 7.0	98.4 ± 1.3
1 to 2×10^{-3}	90.5 ± 3.2	92.6 ± 4.8	93.3 ± 2.8	92.1 ± 3.0	93.3 ± 3.0	93.3 ± 2.8	87.2 ± 3.6	96.0 ± 1.7	78.0 ± 7.2	98.2 ± 1.2
1 to 2×10^{-2}	77.7 ± 4.4	86.6 ± 6.9	79.8 ± 3.6	85.6 ± 3.9	81.2 ± 4.3	87.6 ± 3.1	65.7 ± 6.4	91.9 ± 1.8	66.2 ± 11.1	96.2 ± 1.8

Table 3 - Accuracy on simulated 454 reads.
The gene prediction accuracy (mean and standard deviation over all species in the simulated metagenome) of four metagenomic gene prediction tools GeneMark, MetaGene, MetaGeneAnnotator (MGA) and Orphelia, and of the EST processing tool ESTScan on simulated 454 reads is shown.

	GeneMark		MetaGene		MGA		Orphelia		ESTScan	
Error rate	Sens.[2]	Spec.[3]	Sens.[2]	Spec.[3]	Sens.[2]	Spec.[3]	Sens.[2]	Spec.[3]	Sens.[2]	Spec.[3]
0	91.0 ± 3.6	93.8 ± 4.8	95.4 ± 2.8	92.8 ± 2.4	94.6 ± 2.7	94.1 ± 2.5	88.4 ± 3.5	96.7 ± 1.7	81.3 ± 7.8	97.9 ± 1.4
0.0022	85.3 ± 4.2	90.4 ± 5.6	89.3 ± 3.1	89.2 ± 3.5	89.6 ± 3.3	90.8 ± 2.6	80.0 ± 4.2	94.7 ± 2.1	77.2 ± 9.0	97.2 ± 1.5
0.0049	79.5 ± 4.9	87.6 ± 6.4	83.7 ± 3.5	85.9 ± 3.9	84.7 ± 4.0	87.7 ± 2.8	70.9 ± 5.9	92.5 ± 2.1	71.7 ± 11.5	96.2 ± 1.7
0.028	36.8 ± 4.9	68.3 ± 8.0	39.6 ± 3.9	60.6 ± 8.8	43.3 ± 5.5	61.9 ± 3.6	26.3 ± 9.1	68.3 ± 5.0	26.4 ± 11.2	86.2 ± 4.7

Table 4 - Accuracy by Species

The gene prediction accuracy (mean and standard deviation over all species in the simulated metagenome) of four metagenomic gene prediction tools GeneMark, MetaGene, MetaGeneAnnotator (MGA) and Orphelia, and of the EST processing tool ESTScan on pyrosequencing reads (450 nt) with 0.49% errors.

	GeneMark		MetaGene		MGA		Orphelia		ESTScan	
Species	Sens.[2]	Spec.[3]	Sens.[2]	Spec.[3]	Sens.[2]	Spec.[3]	Sens.[2]	Spec.[3]	Sens.[2]	Spec.[3]
GC-content 30 - 39%										
B. aphidicola	81.7	92.6	88.5	88.1	90.1	88.6	66.0	93.5	71.4	98.3
A. laidlawii	86.2	94.6	90.4	89.9	91.9	91.3	75.3	96.2	81.4	97.3
P. marinus	81.8	91.3	84.0	87.1	84.2	87.9	61.9	92.5	69.3	97.3
D. thermophilum	85.1	94.2	84.7	90.5	86.6	91.2	71.8	94.6	70.5	98.5
N. maritimus	85.7	90.4	86.2	86.7	87.7	87.7	68.1	92.4	76.9	96.1
W. endosymbiont	80.1	81.0	83.6	84.0	84.7	85.6	65.2	90.9	67.6	94.5
Hydrogenobaculum sp.	83.9	93.9	84.3	89.9	85.5	90.8	67.2	93.1	73.0	98.1
GC-content 40 - 49%										
D. kamchatkensis	73.3	94.3	77.1	89.5	79.5	90.4	61.4	93.1	35.0	94.4
E. sibiricum	78.9	90.0	81.7	88.2	81.8	89.4	78.3	93.9	74.4	95.1
GC-content 50 - 59%										
H. aurantiacus	70.2	80.1	80.2	84.9	78.6	86.1	74.0	92.7	76.4	95.5
C. tepidum	74.0	76.6	78.6	81.0	79.0	83.0	72.7	89.9	72.7	93.7
GC-content 60 - 69%										
C. jeikeium	77.2	83.7	84.2	83.3	86.6	88.2	79.7	93.2	76.1	96.3
N. pharaonis	77.0	83.5	83.3	81.6	83.9	84.8	72.8	91.3	75.3	94.4
B. pseudomallei	77.7	80.5	85.2	77.6	85.9	83.3	77.0	87.8	84.5	97.7

Table 5 - Accuracy on FAMeS reads.

The gene prediction accuracy of four metagenomic gene prediction tools GeneMark, MetaGene, MetaGeneAnnotator (MGA) and Orphelia, and of the EST processing tool ESTScan on FAMeS reads [4] is shown.

	simLC[4]		simMC[5]		simHC[6]	
Method	Sens.[2]	Spec.[3]	Sens.[2]	Spec.[3]	Sens.[2]	Spec.[3]
GeneMark	78.8	85.9	77.3	85.1	77.1	83.0
MetaGene	80.0	78.4	78.8	77.5	78.0	74.9
MetaGeneAnnotator	79.6	80.2	78.4	79.4	77.3	75.6
Orphelia	76.7	85.0	74.9	82.5	74.8	82.0
ESTScan	70.2	96.0	69.3	96.1	69.0	95.0

Additional Files

Additional file 1

Supplementary Materials

This pdf-file contains supplementary details to the section Methods, supplementary tables and supplementary figures.

Figure 1

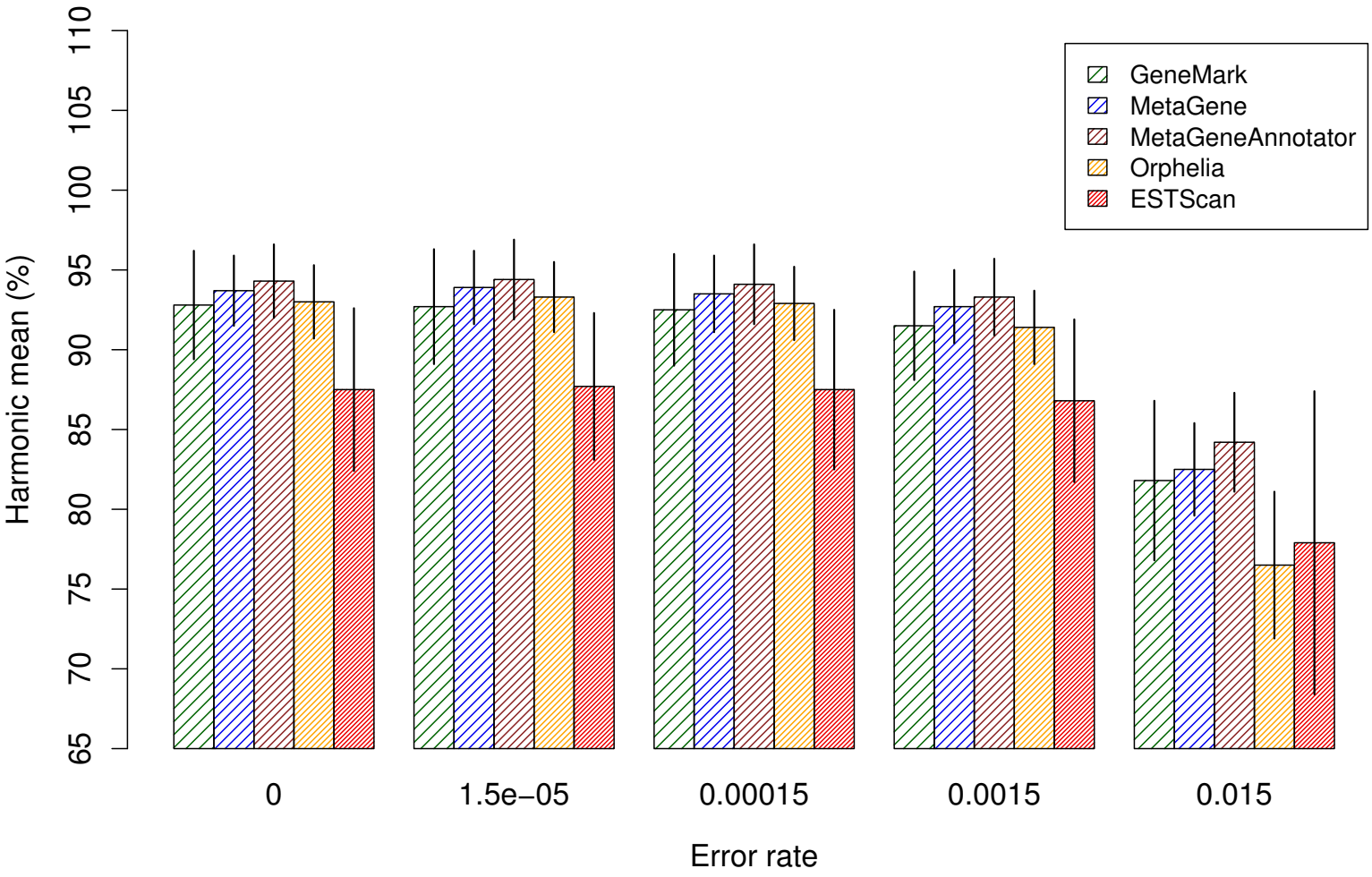
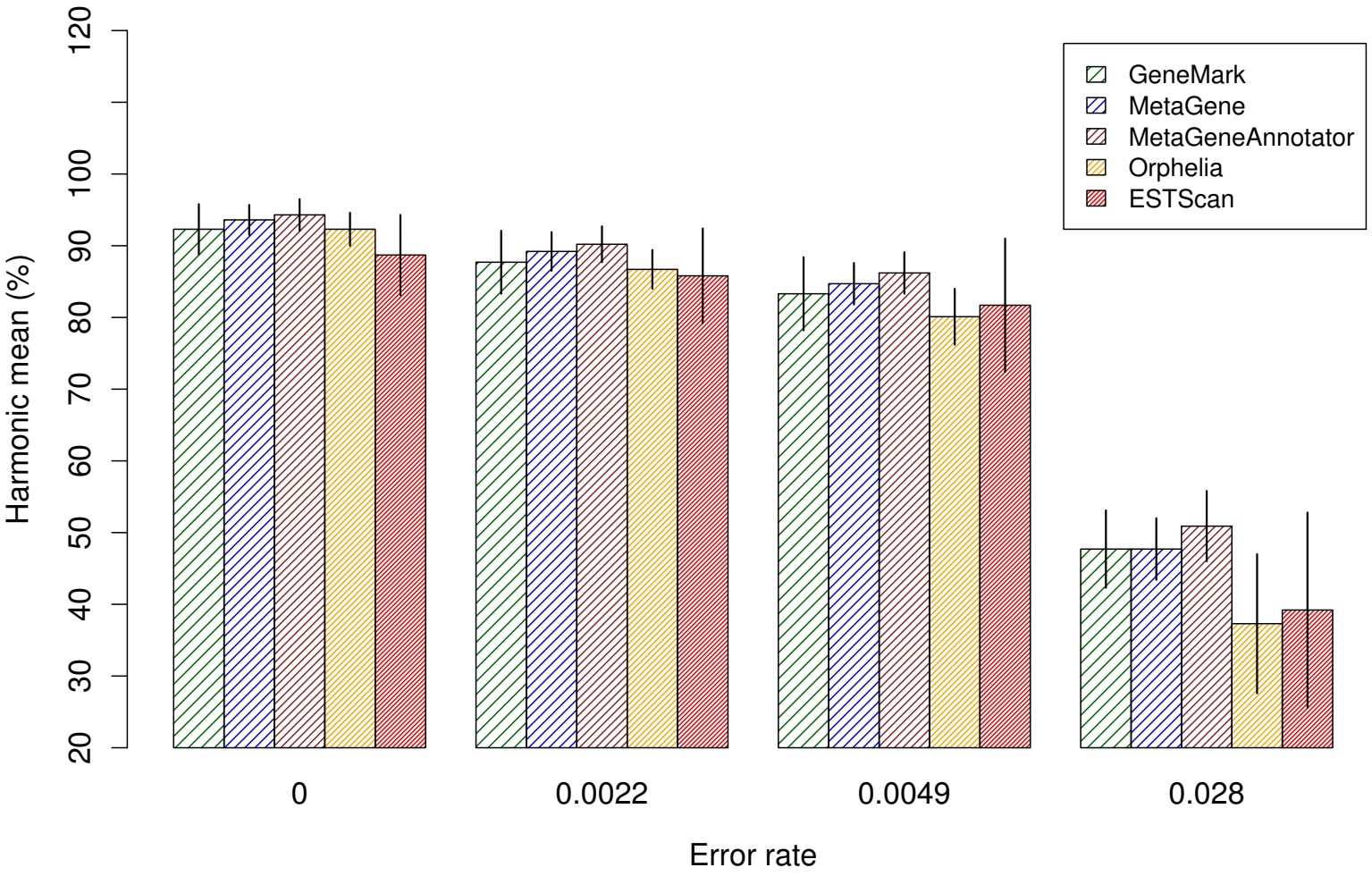

Harmonic mean (%)
65
70
75
80
85
90
95
100
105
110
0
1.5e−05
0.00015
0.0015
0.015
Error rate
GeneMark
MetaGene
MetaGeneAnnotator
Orphelia
ESTScan

Figure 2


Harmonic mean (%)
20
30
40
50
60
70
80
90
100
120
0
0.0022
0.0049
0.028
Error rate
GeneMark
MetaGene
MetaGeneAnnotator
Orphelia
ESTScan

Additional files provided with this submission:

Additional file 1: bmc_supplementary.pdf, 548K
http://www.biomedcentral.com/imedia/1428483089321559/supp1.pdf

Chapter 6

General Discussion

Orphelia is an accurate metagenomic gene prediction tool. In comparison to other metagenomic gene prediction methods, Orphelia has several unique features:

- We showed with Orphelia that the GC-content of an input sequence is not necessary in order to derive a suitable model for scoring the codon usage of this sequence. Instead, one generalized codon usage model and one generalized di-codon usage model are applied to sequences with all GC-contents.
- In contrast to MetaGene and MetaGeneAnnotator, Orphelia also shows that a good gene prediction accuracy can be achieved for *Bacteria* and *Archaea* without utilizing different scoring models for sequences from the two domains.
- Due to its very generalized codon usage models, Orphelia predicts 'typical' genes with a uniquely high specificity. In turn, some 'atypical' genes that more sensitive tools will detect, are overlooked. The question is whether a high reliability of the predictions or the (over-) prediction of genes is more desirable for metagenomic projects. Considering that the manual curation of metagenomic gene predictions is often not feasible, the specificity of a method is probably of higher importance.

Apart from the unique features of Orphelia, several general aspects of current metagenomic gene prediction research need to be discussed. The following sections deal with the selection of appropriate training data, the inclusion of additional genomic features of PCGs into metagenomic gene prediction programs, and with the applicability of metagenomic gene prediction tools to different kinds of real data.

6.1 Training Data

The Orphelia gene prediction tool as presented in chapters 3 and 4 was trained on a set of fully sequenced genomes from 131 species that were initially selected by Noguchi *et al.* to represent every genus from GenBank in the year 2006 [19]. By now, species belonging to many other genera have been fully sequenced and annotated. In addition, many other microbial genome projects in progress could provide valuable training data. Members of these new genera should be included in the training set of a future gene prediction tool version in order to collect as much information about the characteristics of coding and non-coding ORFs as possible.

Another topic is whether the 'one species per genus' criterion is actually suitable for selecting training species. Generally, taxonomy does not reflect phylogeny properly. Some species of different genera for example exhibit highly similar codon usage patterns. In addition, the species whose genomes were sequenced and deposited in public databases, e.g. GenBank, are biased in the sense that most of those species can be kept in culture under laboratory conditions [27]. This phenomenon is called the 'GenBank bias'. Particularly for the identification of novel genes in metagenomes whose biological diversity is yet unknown, the transfer of the GenBank bias toward single species should be avoided in the training data set. To reduce this bias, training genomes could be selected according to other criteria, e.g. GC-content, oligonucleotide frequencies or monocodon/dicodon frequencies in protein coding regions.

6.2 More Features of Protein Coding Genes

Gene prediction algorithms are designed to distinguish between PCGs and non-coding open reading frames (nORFs) using suitable features. In agreement with Noguchi *et al.* (2006), we show in chapter 3 that mono- and dicodon frequencies (illustrated in figure 1.3 on page 8) are useful features. The ORF length is another intuitive feature that is used by our method (chapter 3) and in MetaGene.

Other hints at whether an ORF is a protein coding or not can be derived from signals that are necessary to drive the transcription and translation machinery. Translation initiation signals, which are located in close proximity to the translation start codon, turned out to be very useful, particularly for predicting a gene with its correct start site. This feature has been implemented in Orphelia (chapters 3 and 4). In a slightly different manner, it has also been incorporated into the MetaGeneAnnotator [20]. In theory, other signals, like the transcription start site, or the transcription termination signal could also be useful. Here, we encounter the problem that both types of features are not necessarily located close to

the actual coding region. Signals that are distant from the actual coding region are unlikely to be captured within a metagenomic sequencing read, and thus, the incorporation of such signals is unlikely to improve gene prediction in metagenomic Sanger and pyrosequencing reads.

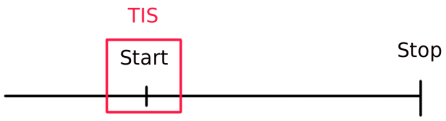


Figure 6.1: The translation initiation site (TIS) of a gene is a useful feature for discriminating true and wrong start codons of a gene, but it also gives further evidence at whether an ORF is protein coding at all. In Orphelia, 60 b window around the start codon is used to extract the TIS feature.

6.3 Applicability to Real Data

All metagenomic gene prediction tools aim to predict PCGs in sequences that originate from probably completely unknown microorganisms with models that were derived from already characterized organisms. In chapters 3 and 4, but also in the publications of MetaGene and MetaGeneAnnotator, it was shown that metagenomic gene prediction models can successfully be applied to reads from unknown species by excluding certain 'test genera' from the 'set of training species'. The evaluation was later carried out on DNA fragments of species that belong to the 'test genera'. This analysis gives a hint at whether the models will work for predicting genes in reads from entirely new species. However, a certain level of uncertainty remains. The experimental verification of predicted genes in metagenomes, e.g. with help of the metatranscriptome, will give further clues, but to date, the required data are not available.

The previously mostly unexplored question, to which extent metagenomic gene prediction tools are affected by sequencing errors, is addressed in chapter 5. The conclusion of this chapter is that all tools in principle work well for Sanger reads, and that the sequencing errors in 454 reads pose a significant problem. However, this does not yet tell whether the tools are applicable or not applicable to all real data sets. Metagenomic sequencing projects differ in their aims and in the applied sequencing strategies. Sanger- and pyrosequencing are frequently used for metagenome sequencing (examples can be found in [7, 8, 28, 9, 29]). However, in some metagenomic sequencing projects, completely different strategies are applied. One example is the sequencing of long metagenomic inserts (up to 40 knt) with multiple coverage, leading to long DNA fragments with low error rates (e.g. demonstrated by Voget *et al.*, 2003 [30]). Such long fragments are in comparison to short reads easily

classified into phylogenetic categories (e.g. using TETRA [31], Phylopythia [17] or TACOA [32]), which makes even the application of conventional gene finders with pre-trained models possible (see section 1.3 on page 6).

With increasing read length and an even cheaper price per base than the 454 system, other next generation sequencing (NGS) techniques than 454, for instance the Illumina sequencing system [33], will become attractive for metagenomics. Every NGS technique has its own, typical kind of sequencing errors and error rates. Therefore, it will be necessary to re-evaluate the effect of sequencing errors on metagenomic gene prediction for each NGS system.

Also in short future, another approach for sequencing very long reads will become available. This is going to be the system of Pacific Biosciences [34]. Those reads are going to be tens of thousands of b long but in contrast to the long-insert sequencing with multiple coverage, they are going to contain sequencing errors. One can speculate that gene prediction techniques that are applicable to Sanger reads might also perform well on those reads but for making definite conclusions, the methods need to be evaluated on data sets that are simulated according to the properties of the new sequencing techniques.

Chapter 7

Summary and Conclusions

Metagenomic sequencing projects generate huge amounts of single read data that cannot be analyzed efficiently with conventional gene prediction tools that were designed for long genomic sequences. Often, a six frame translation of the single reads is searched against databases of known proteins with BLAST but this can only lead to the detection of genes that are highly similar to already known genes. Novel genes, that are to be expected in microbial communities, cannot not be identified.

Two principle approaches have been developed to solve the problem of gene prediction in metagenomic single read data. One is the identification of PCGs by intra and inter metagenome sequence homology. These methods are computationally very expensive. Another approach is the prediction of PCGs with statistical models. These methods are definitely faster by several orders of magnitude.

In this work, the metagenomic gene prediction method Orphelia is introduced. Orphelia consists of a two-stage machine learning approach. In the first stage, linear discriminants for monocodon usage, dicodon usage and translation initiation sites are used to extract features from DNA sequences. In the second stage, an artificial neural network combines these features with open reading frame length and fragment GC-content to compute the probability that this open reading frame encodes a protein. This probability is used for the classification and scoring of gene candidates. In comparison to other model-based metagenomic gene prediction tools, Orphelia has a very high specificity and a slightly lower sensitivity on Sanger and 454 reads. Orphelia was the first tool that incorporated the detection of TIS signals into the gene prediction process, thereby also ensuring the reliable prediction of the correct gene start.

During the investigation of the applicability of metagenomic gene prediction tools to real data, sequencing errors have often been neglected. Here, we show that the effect of typical

Sanger sequencing errors is in most cases minuscule but that pyrosequencing errors can decrease prediction accuracy drastically. We also demonstrate that ESTScan, although not designed for the task at hand, outperforms some metagenomic gene prediction tools on reads with high error rates by its built-in error compensating capabilities.

Overall, Orphelia is demonstrated to be a valuable tool for metagenomic gene prediction in Sanger and 454 reads if a high prediction specificity is desired. This is likely to be the case because the manual curation of predictions in metagenomics is often impossible due to the sheer amount of data. However, the integration of error-compensating methods into Orphelia - and actually all other metagenomic gene prediction tools - is desirable in order to improve the applicability to real 454 data.

Bibliography

[1] T.P. Curtis, W.T. Sloan, and J.W. Scannel. Estimating prokaryotic diversity and its limits. *Proc. Natl. Acad. Sci. USA*, 99(22):10494–10499, 2002.

[2] R.I. Amann, W. Ludwig, and K.H. Schleifer. Phylogenetic identification and in situ detection of individual microbial cells without cultivation. *Microbiol. Rev.*, 59(1):143–169, 1995.

[3] M.S. Rappe and S.L. Giovannoni. The uncultured microbial majority. *Annu. Rev. Microbiol.*, 57:369–394, 2003.

[4] E. Kellenberger. Exploring the unknown: the silent revolution of microbiology. *EMBO Rep.*, 2:5–7, 2001.

[5] F. Sanger, S. Nicklen, and A. R. Coulson. DNA sequencing with chain-terminating inhibitors. *Proc. Natl. Acad. Sci. USA*, 74(12):5463–5467, 1977.

[6] M. Ronaghi, M. Uhlén, and P. Nyreén. A sequencing method based on real-time pyrophosphate. *Science*, 281(5375):363–365, 1998.

[7] M.R. Rondon, P.R. August, A.D. Bettermann, S.F. Brady, T.H. Grossman, M.R. Liles, K.A. Loiacono, B.A. Lynch, I.A. MacNeil, C. Minor, C. Lai Tiong, M. Gilman, M.S. Osburne, J. Clardy, J. Handelsman, and R.M. Goodman. Cloning the soil metagenome: a strategy for accessing the genetic and functional diversity of uncultured microorganisms. *Appl. Env. Microbiol.*, 66(6):2541–2547, 2000.

[8] J.C. Venter, K. Remington, J.F. Heidelberg, A.L. Halpern, D. Rusch, J.A. Eisen, D. Wu, I. Paulsen, K.E. Nelson, W. Nelson, D.E. Fouts, S. Levy, A.H. Knap, M.W. Lomas, K. Nealson, O. White, J. Peterson, J. Hoffman, R. Parsons, H. Baden-Tillson, C. Pfannkoch, Y. H. Rogers, and H.O. Smith. Environmental shotgun sequencing of the sargasso sea. *Science*, 304:66–74, 2004.

[9] V. Kunin, J. Raes, J.K. Harris, J.R. Spear, J.J. Walker, N. Ivanova, C. von Mering, B.M. Bebout, N.R. Pace, P. Bork, and P. Hugenholtz. Millimeter-scale genetic gradients

and community level molecular convergence in a hypersaline microbial mat. *Molecular Systems Biol.*, 4:198, 2008.

[10] K. Mavromatis, N. Ivanova, K. Barry, H. Shapiro, E. Goltsman, A.C. McHardy, I. Rigoutsos, A. Salamov, F. Korzeniewski, M. Land, A. Lapidus, I. Grigoriev, P. Richardson, P. Hugenholtz, and N.C. Kyrpides. Use of simulated data sets to evaluate the fidelity of metagenomic processing methods. *Nat. Meth.*, 4(6):1548–7091, 2007.

[11] S.F. Altschul, W. Gish, W. Miller, E.W. Myers, and D.J. Lipman. Basic local alignment search tool. *J. Mol. Biol.*, 215:403–410, 1990.

[12] L. Krause, N. N. Diaz, D. Bartels, R. A. Edwards, A. Pühler, F. Rohwer, F. Meyer, and J. Stoye. Finding novel genes in bacterial communities isolated from the environment. *Bioinformatics*, 22(14):e281–e289, 2006.

[13] S. Yooseph, G. Sutton, D.B. Rusch, A.L. Halpern, S.J. Williamson, K. Remington, J.A. Eisen, K.B. Heidelberg, G. Manning, W. Li, L. Jaroszewski, P. Cieplak, C.S. Miller, H. Li, S.T. Mashiyama, M.P. Joachimiak, C. van Belle, J.-M. Chandonia, D. A. Soergel, Y. Zhai, K. Natarajan, S. Lee, B.J. Raphael, V. Bafna, R. Friedmann, S. E. Brenner, A. Godzik, D. Eisenberg, J.E. Dixon, S.S. Taylor, R.L. Strausber gand M. Frazier, and J.C. Venter. The Sorcerer II global ocean sampling expedition: Expanding the universe of protein families. *PLoS Biology*, 5(3):0432–0466, 2007.

[14] W. Li and A. Godzik. Cd-hit: a fast program for clustering and comparing large sets of protein or nucleotide sequences. *Bioinformatics*, 22:1658–9, 2006.

[15] A. L. Delcher, D. Harmon, S. Kasif, O. White, and S. L. Salzberg. Improved microbial gene identification with GLIMMER. *Nucleic Acids Res.*, 27(23):4636–4641, 1999.

[16] A. Lukashin and M. Borodovsky. GeneMark.hmm: new solutions for gene finding. *Nucleic Acids Res.*, 26(4):1107–1115, 1998.

[17] A.C. McHardy, H.G. Martín, A. Tsirigos, P. Hugenholtz, and I. Rigoutos. Accurate phylogenetic classification of variable-length DNA fragments. *Nat. Meth.*, 4(1):63–72, 2007.

[18] J. Besemer and M. Borodovsky. Heuristic approach to deriving models for gene finding. *Nucleic Acids Res.*, 27(19):3911–3920, 1999.

[19] H. Noguchi, J. Park, and T. Takagi. MetaGene: prokaryotic gene finding from environmental shotgun sequences. *Nucleic Acids Res.*, 34(19):5623–5630, 2006.

[20] H. Noguchi, T. Taniguchi, and T. Itoh. MetaGeneAnnotator: Detecting species-specific patterns of ribosomal binding site for precise gene prediction in anonymous prokaryotic and phage genomes. *DNA Res.*, 15(6):387–396, 2008.

[21] M. Borodovsky and D. McIninch. GeneMark: Parallel gene recognition for both DNA strands. *J. Comput. Biochem.*, 17:123–133, 1993.

[22] J. Shendure and H. Ji. Next-generation DNA sequencing. *Nature Biotechnology*, 26(10):1135–1144, 2008.

[23] B. Ewing, H. LaDeana, M. C. Wendl, and P. Green. Base-calling of automated sequencer traces using Phred I. accuracy assessment. *Genome Res.*, 8:175–185, 1998.

[24] M. Margulies, M. Egholm, W. E. Altman, S. Attiya, J.S. Bader, L.A. Bemben, J. Berka, M.S. Braverman, Y.-J. Chen, Z. Chen, S.B. Dewell, L. Du, J. M. Fierro, X.V. Gomes, B. C. Godwin, W. He, S. Helgesen, C.H. Ho, G.P. Irzyk, S.C. Jando, M.L.I. Alenquer, T.P. Jarvie, K.B. Jirage, J.-B. Kim, J.R. Knight, J.R. Lanza, J.H. Leamon, S.M. Lefkowitz, M. Lei, J. Li, K.L. Lohman, H. Lu, V.B. Makhijani, K.E. McDade, M.P. McKenna, E.W. Myers, E. Nickerson, J.R. Nobile, R. Plant, B.P. Puc, M.T. Ronan, G.T. Roth, G.J. Sarkis, J.F. Simons, J.W. Simpson, M. Srinivasan, K.R. Tartaro, A. Tomasz, K.A. Vogt, G.A. Volkmer, S.H. Wang, Y. Wang, M.P. Weiner, P. Yu, R.F. Begley, and J.M. Rothberg. Genome sequencing in microfabricated high-density picolitre reactors. *Nature*, 437:376–380, 2005.

[25] W. Brockman, P. Alvarez, S. Young, M. Garber, G. Giannouskos, W.L. Lee, C. Russ, E.S. Lander, C. Nusbaum, and D.B. Jaffe. Quality scores and SNP detection in sequencing-by-synthesis systems. *Genome Res.*, 18:763–770, 2008.

[26] M. Tech. *Analyse von Translationsstarts in prokaryotischen Genomen mit Methoden des Maschinellen Lernens*. PhD thesis, Georg-August-Universität Göttingen, 2007.

[27] P. Hugenholtz. Exploring prokaryotic diversity in the genomic era. *Genome Biol.*, 3(2):reviews0003.1–0003.8., 2002.

[28] G.W. Tyson, J. Chapman, P. Hugenholtz, E.E. Allen, R.J. Ram, P.M. Richardson, V.V. Solovyev, E.M. Rubin, D.S. Rokhsar, and J.F. Banfield. Community structure and metabolism through reconstruction of microbial genomes from the environment. *Nature*, 428:37–43, 2004.

[29] R. A. Edwards, B. Rodriguez-Britol, L. Wegley, M. Haynes, M. Breitbart, D. M. Peterson, M. O. Saar, S. Alexander, E. C. Alexander, and F. Rohwer. Using pyrosequencing to shed light on deep mine microbial ecology. *BMC Genomics*, 7(57), 2006.

[30] S. Voget, C. Leggewie, A. Uesbeck, C. Raasch, K.-E. Jaeger, and W. R. Streit. Prospecting for novel biocatalysts in a soil metagenome. *Appl. Env. Microbiol.*, 69(10):6235–6242, 2003.

[31] H. Teeling, J. Waldmann, T. Lombardot, M. Bauer, and F.O. Glöckner. TETRA: a web-service and a stand-alone program for the analysis and comparison of tetranucleotide usage patterns in DNA sequences. *BMC Bioinformatics*, 5:163, 2004.

[32] N.N. Diaz, L. Krause, A. Goesmann, K. Niehaus, and T.W. Nattkemper. TACOA – taxonomic classification of environmental genomic fragments using a kernelized nearest neighbor approach. *BMC Bioinformatics*, 10:56, 2009.

[33] Illumina, New York, NY. *Protocol for Whole Genome Sequencing using Solexa Technology. In: BioTechniques Protocol Guide*, 2006.

[34] J. Eid, A. Fehr, J. Gray, K. Lyong, J. Lyle, G. Otto, P. Peluso, D. Rank, P. Baybayan, B. Bettman, A. Bibillo, K. Bjornson, B. Chaudhuri, F. Christians, R. Cicero, S. Clark, R. Dalal, A. deWinter, J. Dixon, M. Foquet, A. Gaertner, P. Hardenbol, C. Heiner, K. Hester, D. Holden, G. Kearns, X. Hong, R. Kuse, Y. Lacroix, S. Lin, P. Lunquist, C. Ma, P. Marks, M. Maxham, D. Murphy, I. Park, T. Pham, M. Phillips, J. Roy, R. Sebra, G. Shen, J. Sorenson, A. Tomaney, K. Travers, M. Trulson, J. Vieceli, J. Wegener, D. Wu, A. Yang, D. Zaccarin, P. Zhao, F. Zhong, J. Korlach, and S. Turner. Real-time DNA sequencing from single polymerase molecules. *Science*, 323:133–138, 2009.

Curriculum Vitae

Personal data

Name:	Katharina Jasmin Hoff
Date of birth:	07.04.1983
Place of birth:	Sieburg, Germany
Nationality:	German

Schooling

09/1989 - 07/1990	Adelheidisgrundschule Bonn (elementary school)
08/1990 - 07/1993	Grundschule Birk (elementary school)
09/1993 - 01/1995	Gymnasium Lohmar
02/1995 - 03/2002	Stefan-George-Gymnasium in Bingen am Rhein Abitur Majors: Biology, German and English

Studies

10/2002 - 09/2005	Plant Biotechnology Leibniz Universität Hannover Bachelor of Science Thesis: "R-Manual for Students of Horticulture and Plantbiotechnology"
since 10/2005	Molecular Biology Georg-August-Universität Göttingen MSc/PhD studies Thesis: "Gene Prediction in Metagenomic Sequencing Reads"

Stipends

07/2003 - 09/2005	Stiftung der Deutschen Wirtschaft stipend
10/2003 - 08/2009	Online-stipend by e-fellows.net
07/2004 - 06/2005	ERASMUS stipend
10/2005 - 09/2006	Max Planck Research School stipend
10/2006 - 12/2008	Georg-Christoph-Lichtenberg stipend

Work Experience

04/2003 - 07/2004	Student assistant Leibniz Universität Hannover organisation of the evaluation of teaching
10/2007 - 01/2009	Graduate assistant Georg-August-Universität Göttingen teaching a statistics course with the language R
since 01/2009	Scientific assistant Georg-August-Universität Göttingen