Combination of results from gene-finding programs

Cecilia Hammar

Submitted by Cecilia Hammar to the University of Skövde as a dissertation towards the degree of M.Sc. by examination and dissertation in the Department of Computer Science.

September 1999

I hereby certify that all material in this dissertation which is not my own work has been identified and that no work is included for which a degree has already been conferred on me.

Cecilia Hammar

HS-IDA-MD-99-010

Abstract

Gene-finding programs available over the Internet today are shown to be nothing more than guides to possible coding regions in the DNA. The programs often do incorrect predictions. The idea of combining a number of different gene-finding programs arised a couple of years ago. Murakami and Takagi (1998) published one of the first attempts to combine results from gene-finding programs built on different techniques (e.g. artificial neural networks and hidden Markov models). The simple combinations methods used by Murakami and Takagi (1998) indicated that the prediction accuracy could be improved by a combination of programs.

In this project artificial neural networks are used to combine the results of the three well-known gene-finding programs GRAILII, FEXH, and GENSCAN. The results show a considerable increase in prediction accuracy compared to the best performing single program GENSCAN.

Contents

1 Introduction

Since the mid 1990's the genome projects around the world have discovered a large number of deoxyribonucleic acid (DNA) sequences from various species. The DNA stores the genetic information, i.e. the genes, of all living creatures (except RNA viruses). Genes are regions of DNA that code for proteins. It is only small part of the mammalian DNA that is part of some gene. The larger part of the DNA does not appear to code for any protein. The genes are divided in small parts called exons with surrounding regions of non-coding DNA called introns see Chapter 2.1.

Finding genes in DNA is a difficult task and there is a great need for computer support in the process. Many different techniques have been used to the gene-searching problem, e.g. artificial neural networks (ANN), hidden Markov models, and Rule Based systems (see Chapter 2.2).

The gene-finding programs that have been developed over the years can be used for pinpointing regions in the DNA that are likely to contain exons. As Burset and Guigó (1996) state, the gene-finding programs are far from being powerful enough to eludicate the genomic structure completely. The performance of some of the most widely used gene-finding programs is shown by a number published performance tests (e.g. Burset and Guigó 1996, Murakami and Takagi, 1998).

Recently the possibilities of combining results from different gene-finding programs have been discussed. There are a number of researchers suggesting that a combination of the results from several programs would give a more reliable result than any single program (e.g. Murakami and Takagi, 1998). Murakami and Takagi (1998) tested four different well-known gene-finding programs (FEXH, GRAILII, GENSCAN, and GeneParser) and evaluated five different methods for combination of the results. The five methods used in that project were called AND, OR, HIGHEST, RULE and BOUNDARY. The AND method simply stated that the regions predicted as coding by all programs were actually coding. This method resulted in the lowest rate of incorrectly predicted exons. The OR method stated that the regions predicted as coding by at least one program were the actual exons. This method resulted in the lowest rate of missing exons. With the HIGHEST method the regions with the highest probability among the programs were the overall predicted exons. The RULE method used a priority order for the programs and the BOUNDARY method used more biological information to make the overall exon prediction. One of the most widely used measures for the association between prediction and reality is the *approximate correlation*. The results of Murakami and Takagi's (1998) study demonstrated an improved approximate correlation (AC) by 3-5% when using the methods HIGHEST and BOUNDARY. The results were compared to the best performing program (GENSCAN) when it was used separately. Murakami and Takagi (1998) also showed that by three of the methods (HIGHEST, OR, and BOUNDARY) the AC improved as the number of programs combined increased.

In this project three new methods for combination of the results from gene-finding programs are evaluated. The AND and OR methods used by Murakami and Takagi (1998) are also evaluated for comparison. Two of the new methods are based on artificial neural networks, while the third is influenced by the logical methods AND and OR. The results will show if there are combination methods that can improve the approximate correlation more than the methods used by Murakami and Takagi (1998).

1.1 Thesis statement

The problem considered in this project is how results from different gene-finding programs should be combined in order to gain the most reliable predictions.

The aim of this project is to evaluate a number of methods for combination of the results from different gene-finding programs and find the best one among them. The aim makes it possible to formulate the following hypothesis which is addressed in this thesis:

An artificial neural network (ANN) will show results that are better than the logical methods AND, OR, and MAJORITY[1].

An ANN can approximate simple AND, OR, and MAJORITY functions (Russell and Norvig, 1995). If in fact one of the methods AND, OR, or MAJORITY gives the best result, the ANN should give equivalent results. If the ANN gives better results than the logical methods then the hypothesis will hold. The results in this project will also imply if there is a more appropriate function (than AND, OR, or MAJORITY) that can be approximated by the ANN.

The hypotheses will be falsified if no ANN shows as good as, or better results than the methods AND, OR, and MAJORITY. This is true if it is proven to be impossible to train the ANN to a satisfying performance on the test set.

The main difference between the experiments in this project and the experiments carried out by Murakami and Takagi (1998) is how the prediction scores and weights are used. In

[1] MAJORITY can be decomposed into simple logical AND and OR operator.

this project the scores and weights associated with program predictions will be used normalized. In the experiments performed by Murakami and Takagi (1998) the scores associated with program prediction were transformed into probabilistic scores.

The data set collected form GenBank by Murakami and Takagi (1998) will be used in this project. The sequences in the data set are shown in appendix A and the criteria for sequence selection used by Murakami and Takagi (1998) are presented in Chapter 3.2.

1.2 The project objectives

There are four main objectives identified in this project that together will meet the aim and fulfil or falsify the hypotheses. The objectives are:

- Three gene-finding programs will be selected and used on the data set. The same programs that were used by Murakami and Takagi (1998) will be used (i.e. GRAILII, GENSCAN, and FEXH) with an exception for GeneParser that cannot be used due to hardware compatibility problems. The data set collected from GenBank by Murakami and Takagi (1998) will be used since the criteria and motivations are well documented. The results will be analyzed.

  The results from GRAILII, GENSCAN, and FEXH are needed as input to the combination methods evaluated in this project. For evaluation of the combination methods the results of the separate programs will be used.

- Two of the combination methods (AND, OR) used by Murakami and Takagi (1998) and a new logical method (MAJORITY) will be used to combine the results from the three gene-finding programs.

  The results of the AND, OR, and MAJORITY methods will be compared to the results of the separate programs and they will be used for comparison with the ANNs.

- A machine learning method, ANN, will be used for combination of results. The results will be compared to the results from the previous tests done in this project. If the ANN is able to learn how to combine the predictions of the programs the hypothesis will be fulfilled.

- An ANN using a sliding window approach will be used and results will be compared with the other tests performed in the project. The sliding window ANN has more knowledge about the surrounding nucleotides. Results from the sliding window ANN will be additional to the basic ANN if the basic ANN fulfills the hypothesis.

1.3 The motivation for this project

The accuracy of the gene-finding programs that are available over the Internet today is not satisfactory. Analysis of a DNA sequence with these programs would only give a hint of where the interesting regions are (i.e. possible exons). Many different techniques have been tried on the gene-finding problem. None of the techniques have increased the accuracy considerably. Recently the idea of combining different techniques has been discussed. A number of researchers (see Chapter 2.4) have discussed the possibility and advantages of combining the results from different gene-finding programs. Murakami and

Takagi (1998) showed that the average accuracy was generally improved when combining programs. The results of the combinations of programs were compared to the results produced by the programs when used separately. Murakami and Takagi (1998) claim that the accuracy can be improved considerably if the right combination method is found.

The main motivation for this project is to evaluate methods for combination of gene-prediction programs. The methods used here are based on two very different approaches to the combination problem. The AND, OR, and MAJORITY (MAJ) methods are logical methods with discrete input, while the ANN has the possibility of learning the situations for which there are exceptions from the logical approach. In other words the ANN will approximate a more appropriate function for the combination of the results if there is one. If AND, OR or MAJ is the best combination method the ANN should perform somewhat the same. The results of this project will show which approach is the most promising for combination of predictions.

The problem of combining predictions can be found in a variety of situations. Whenever there are two or more predictions (possibly different) for the same problem, there is the problem of knowing which to trust. It is possible that a combination of predictions can give the most reliable result. A small example: The situation is a line of people walking through Customs. Everybody is walking through the green gate and they have “nothing to declare”. Of course even drug smugglers walk through the green gate. At the gate there is a customs officer, a German shepherd, and a computer with camera device (built and trained using some machine learning technique), all placed there to reveal the smugglers. In situations when the officer, the dog, and the computer predict that a person is a potential smuggler, it is almost certainly true. The problem arises in situations when one or two of the ‘systems’ predict that a person is a smuggler. In certain situations the dog should be trusted and not the officer and so on. The problem is how to combine the predictions.

The results from this project will show the differences between the chosen methods and which combination method is the most appropriate one for the combination of gene predictions.

Another more general motivation for the project is the effort of gaining better and better accuracy in gene-searching. The results in this project will show which combination of programs is the best among all possible combinations of all programs. The best method among the ones tested in this project) for combination of the gene-predictions will be shown. The results in this project will also show which of the three programs is the most accurate one on the data set used here. An increase in prediction accuracy is expected by most of the combination methods compared to when the programs are used separately.

1.3 Overview of thesis

The necessary biological background for understanding the problem is presented in Chapter 2. The different gene-finding programs used in this project are described. The section also contains a short review of work done in the area of combining results from gene-finding programs. Experiments that are carried out in this project are presented in Chapter 3. The results from the experiments are presented in Chapter 4. The analysis of the results is found in Chapter 5. The analysis is focused on the difference in performance between the combination methods and the single programs. The affect on the combination methods by the number of programs combined is also considered. In Chapter 6 a discussion is held to pinpoint the details in the project that affect the results. Finally in Chapter 7 the conclusions that can be drawn from the results in this project are presented.

2 Background

The problem addressed in this project is within the bioinformatics science. The problem is computational, while the problem domain is within molecular biology. This chapter aims at introducing the background for the problem addressed in this project. The basics in molecular biology are very briefly described as well as the three gene-finding programs used in this project. The discussion by researchers that have addressed the idea of combining results from gene-finding programs is presented.

It is not necessary to know the whole biological background to understand the problem. The problem of combining prediction results can be found in many situations and in other fields of science. Predictions are more or less accurate. Intuitively, a combination of several predictions for the same problem might increase the certainty that the prediction is correct if the predictions are similar. The problem is when the predictions are different for the same problem. Which prediction is the most reliable? Is it different in different situations? The problem of combining predictions is considered in this project.

Simple methods for combination of the predictions could be the AND or OR methods used by Murakami and Takagi (1998). The hypothesis of this project is that a machine learning method like an ANN would perform better than the logical method for the problem of combining gene predictions.

2.1 Biological background

Organisms on earth are divided in two groups, *eukaryotes* and *procaryotes*, depending on their cell structure. Eukaryotes are those organisms which have a membrane-bound nucleus (e.g. animals, plants, and fungi) while procaryotes (e.g. bacteria) have a simpler cell structure without nucleus or other membrane-bound structures (Kleinsmith, 1995). As made clear in the introduction, the data used in this project will only consist of human DNA sequences and therefore the focus in the subsequent chapters is on the eukaryotic cell.

The human body consists of about 100 trillion cells (Yap et al. 1996), each of which contains 46 chromosomes in the nucleus. The chromosomes are built from DNA (Deoxyribonucleic acid) and contain about 100,000 genes. A gene is a unit of genetic information (Lewin, 1998). The genes consist of thousands of nucleotide pairs which contain the information needed to specify (i.e. code for) particular proteins.

DNA is a linear polymer consisting of the combinations of four different nucleotide bases (here often referred to as bases); deoxyAdenosine monophosphate (abbreviated Adenine or A), deoxyThymidine monophosphate (Thymine or T), deoxyGuanosine monophosphate (Guanine or G) and deoxyCytidine monophosphate (Cytosine or C). DNA sequences do not exist as single sequences of nucleotides (i.e. strands). The strands are wound around each other in opposite directions (i.e. forward and reverse strand) forming a double helix structure held together by hydrogen bounds (Figure 1). All Gs on one strand are paired with Cs on the other strand, while all As are paired with Ts on the opposite strand. This pattern results in an equal amount of A and T, and an equal amount of G and C. (Lewin, 1998, Griffiths et. al., 1996).

As also shown in Figure 1 the DNA strands have directionality. One end of a strand is called the 5’ while the other end is called the 3’end. The two strands in the DNA double helix run in the opposite directions. One chain runs in the 5’→ 3’ direction while the other runs in the 3’→ 5’ direction. More detailed facts about the DNA double helix can be found in (Kleinsmith and Kish, 1995).

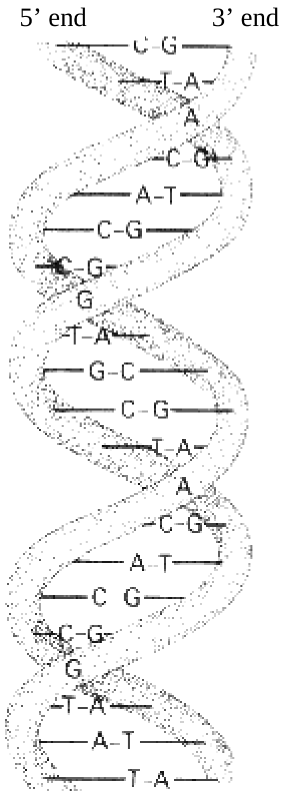


Figure 1. A stylized representation of the DNA double helix. The two DNA strands are held together by hydrogen bounds between Adenine (A) and thymine (T), and between guanine (G) and cytosine (C). The two chains run in opposite directions. One chain runs in the 5’→ 3’ direction while the other runs in the 3’→ 5’ direction.

The relationship between the DNA sequence and the corresponding protein sequence is referred to as the *Genetic Code* (Lewin, 1997).

The human DNA consists of genes with *intergenic regions* in between. These intergenic regions are on average 25-30 kb^2. The genes consist of a number of coding regions (i.e. *exons*) with non-coding regions (i.e. *introns*) in between. This is shown in Figure 2. The number of exons in a gene is generally correlated with the gene length while the exon length is 170 bases on average and independent of the length of the gene. The length of the introns varies enormously and it seems to have a direct correlation with the length of the gene. These facts are found in Strachan and Read (1996), where more details of the human molecular genetics also can be found. In general the exon/intron boundries are crusial to exon prediction. A basic rule that is used to find exons intron boundries is called the GT-AG-rule, which states that introns virtially always start with GT and end with AG. More about intron exon boundries can be foun in Strachan and Read (1996).

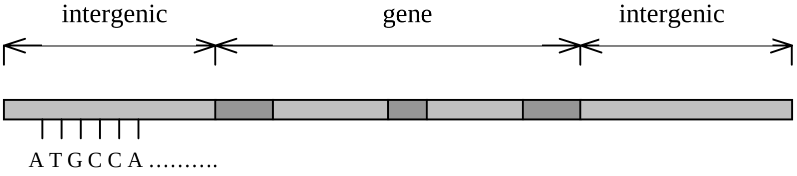


Figure 2. A DNA sequence. Dark gray regions represent exons and light gray regions represent introns.

The *transcription process* that describes the transcription of a gene is shown schematically in Figure 3. Inside the eukaryotic nucleus some genes are active almost constantly while others have to be turned on and off by specific signals from outside the cell (e.g. some hormone substance) or inside the cell (e.g. a signal from a regulatory gene which turns on and off other genes). These signals bind to specific regions of the gene and initiate the *RNA synthesis* that makes a copy of the gene’s DNA. In any given region of the DNA sequence only one of the two strands codes for a protein. The copies are single stranded molecules called RNA. RNA is a polymer similar to DNA consisting of Adenosine monophosphate (A), Guanosine monophosphate (G), Cytidine monophosphate (C) and Uridine monophosphate (which is abbreviated U and is functionally equivalent to Thymidine monophosphate) (Griffiths et.al. 1996). The RNA sequence is complementary to the non-coding strand and identical with the coding strand (except that U is used instead of T).

In the process called *RNA splicing* all introns in the RNA sequence are removed and the resulting mRNA (messenger RNA, transfer RNA or ribosomal RNA) sequence does only contain the coding information (i.e. the exons).

Next, the *translation* process converts the mRNA nucleotide sequence into amino acid sequence building a protein. The nucleotides in the mRNA sequence are read in triplets (codons), each translated into one amino acid.

[2] Kb. Kilo bases. One base is one nucleic acid.

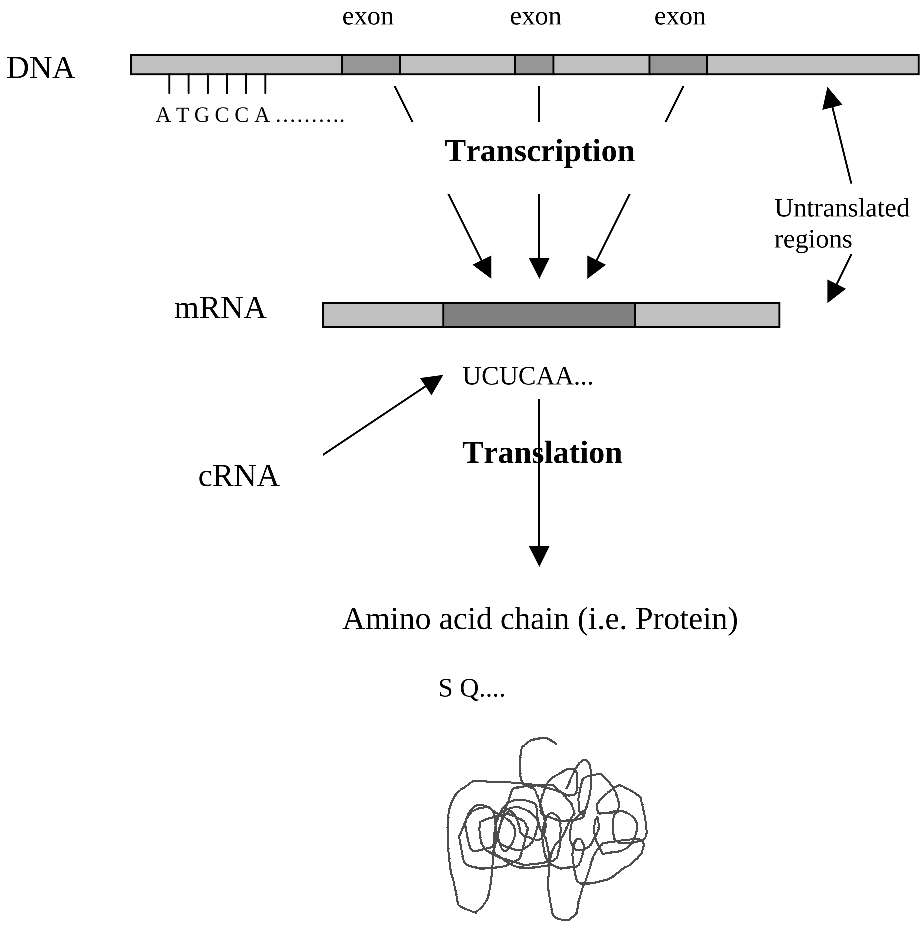


Figure 3. Transcription from DNA to mRNA, and translation from cRNA to amino acids. The amino acid chain folds up into a protein.

2.2 Gene-finding programs

The laboratory analysis of DNA sequences is difficult, time consuming, and expensive which makes computational techniques essential. The complexity of genes (which is not completely understood) makes hand-coded algorithms impractical (Craven and Shavlink, 1994).

Since the 1980s a number of computational methods and approaches have been used to predict protein-coding regions (exons) in genomic DNA sequences. In the early 1990s (Fields and Soderlund, 1990) developed a program, *gm*, which assembled the coding regions in C. elegans DNA into translatable mRNA sequences. Since then, many different computer based techniques have been tried in this domain; artificial neural networks, hidden Markov models (Henderson et. al. 1997) (Krogh, 1997)(Kulp et. al., 1996) (Kulp and Haussler, 1997), dynamic programming (Salzberg, 1997), dynamic programming and neural networks combined (Snyder and Stormo, 1992) (Snyder and Stormo, 1995). Methods based on homology detection in amino acid sequence databases have also been tried (Rogozin et. al., 1998). Salzberg et. al. (1996) used decision trees and dynamic programming . A method based on a prediction algorithm that uses the quadratic discriminant function for multivariate statistical pattern recognition was used by Zhang (1997).

In this project the three well-known gene-finding programs GRAILII, FEXH, and GENSCAN will be used. The program results will be evaluated separately as well as the combinations of the results.

In the following subchapters (2.5.1, 2.5.2, and 2.5.3) the human DNA sequence D49493 will be used for examples. This sequence is part of the data set used in this project and it is found in GenBank. In Chapter 2.6 GenBank is discussed and the actual structure of D49493 will be shown as the example output from the database. The example sequence used in the three subchapters will illustrate the problem of combining the results from gene-finding programs.

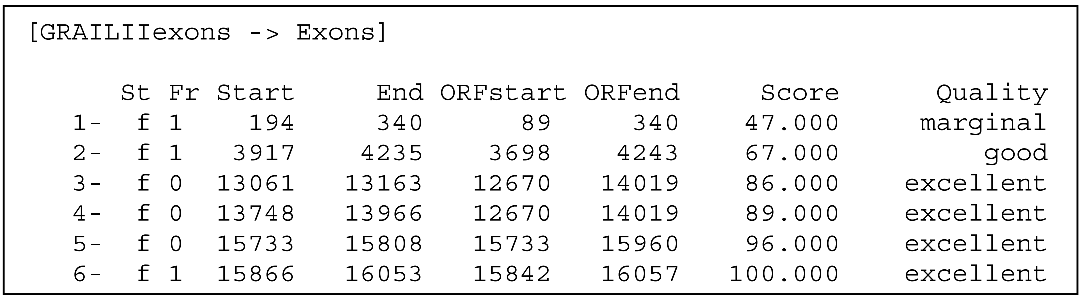
2.2.1 GRAILII

The GRAILII program can be accessed over the web when manually pasting in sequences or through the GRAIL email-server [3]. The resulting GRAILII prediction consists of a set of non-overlapping exons in both forward and reverse strand with assigned scores between 0 and 100 to illustrate the quality of the prediction. In this project only the forward strand predictions are considered since not all programs give a prediction on the reverse strand.

GRAILII uses a neural network which recognizes coding regions within variable-size windows tailored to each potential coding region candidate, defined as an open reading frame bounded by a pair of translation start/donor, acceptor/donor, acceptor/translation stop, or translation start/stop sites. The scheme facilitates the use of genomic context information. For more information about GRAILII see Uberbacher and Mural (1991), Uberbacher et. al. (1993), and Mural et.al. (1992).

An example of a GRAILII exon prediction is shown in Figure 4. *St* is the strand the exon is predicted on (*f* for forward and *r* for reverse). As mentioned earlier, it is only the forward strand predictions that are considered in this project. *Fr* is the reading frame. *Start* and *End* give the start and end positions (in the sequence) of the predicted exon. There is also a Score associated with the predicted exon which reflects the quality of the prediction. The scores are divided in three quality groups; *marginal*, *good* and *excellent*.

[3] GRAILII e-mail server. GRAIL@ornl.gov (with the option “-2” to indicate GRAILII)



```
[GRAILIIexons -> Exons]

     St Fr Start     End ORFstart ORFend     Score       Quality
  1-  f 1    194     340       89    340    47.000      marginal
  2-  f 1   3917    4235     3698   4243    67.000          good
  3-  f 0  13061   13163    12670  14019    86.000     excellent
  4-  f 0  13748   13966    12670  14019    89.000     excellent
  5-  f 0  15733   15808    15733  15960    96.000     excellent
  6-  f 1  15866   16053    15842  16057   100.000     excellent
```

Figure 4. GRAILII prediction for human DNA sequence D49493. The start and end positions of the predicted exons are given together with a score and quality measure of the prediction. In this particular example there are only forward strand predictions. GRAILII also give reverse strand predictions.

The scores associated with the predictions are normalized between 0.2 and 0.8. 0.2 is a non-predicted nucleotide, while 0.8 corresponds to a nucleotide predicted positive with a score of 100.000. The normalization is explained in Figure 8.

2.2.2 GENSCAN

GENSCAN is developed at the Department of Mathematics at Stanford University California USA[4]. It is available for academic users over the web.

GENSCAN is a gene-finding program build to analyze DNA sequences from a number of organisms including human. The program is based on a hidden Makrov model.

[4] GENSCAN. http://bioweb.pasteur.fr/seqanal/interfaces/genscan.html

Using a probabilistic model of the gene's structural and compositional properties of the genomic DNA for a given organism the program determines the most likely gene structure. The probabilistic model used considers many of the essential gene structural properties of DNA sequences e.g. the typical number of exons per gene, the typical gene density, and the distribution of exon sizes for different types of exons (Burge and Karlin, 1997, Burge 1997).

In Figure 5 the GENSCAN prediction of human DNA sequence D49493 is shown. GENSCAN gives the *Begin* and *End* positions for the predicted exons. The *Type* of the predicted exon is also given (Init = Initial exon, Intr = Internal exon, Term = Terminal exon, Sngl = Single-exon gene, Prom = Promoter, PlyA = poly-A signal). The DNA strand on which the predicted exon is located is specified under *S* (+ for input strand, - for opposite strand). The length (i.e. number of base pairs) of the exon is given under *Len*. There are a number of other informative measures (i.e. Fr Ph I/Ac Do/T CodRg) that give even more information of the predicted exons but these variables are not considered in this project. Finally GENSCAN gives a quantity (*P*) that can be considered as the probability of the prediction being correct. The score of the exon (*Tscr*) is dependent on the Len, I/Ac, Do/T and CodRg scores and is used as an overall measure of the prediction quality.

```
GENSCAN 1.0          Date run: 20-May-99   Time: 10:53:17
Sequence D49493 : 17286 bp : 51.57% C+G : Isochore 3 (51 - 57 C+G%)
Parameter matrix: HumanIso.smat
Predicted genes/exons:

Gn.Ex Type S .Begin ...End .Len Fr Ph I/Ac Do/T CodRg P.... Tscr..
 1.01 Init +   3917   4235  319  1  1   94  106   453 0.993  43.21
 1.02 Intr +  13061  13986  926  0  2  108   97  1152 0.993 109.74
 1.03 Intr +  15733  15816   84  0  0   92   55    59 0.705   3.51
 1.04 Intr +  15866  16053  188  1  2  142   97   229 0.552  28.21

Predicted peptide sequence(s):
>D49493|GENSCAN_predicted_peptide_1|506_aa
MAHVPARTSPGPGPQLLLLLLPLFLLLLRDVAGSHRAPAWSALPAAADGLQGDRDLQRHP
GDAAATLGPSAQDMVAVHMHRLYEKYSRQGARPGGGNTVRSFRARLEVVDQKAVYFFNLT
SMQDSEMILTATFHFYSEPPRWPRALEVLCKPRAKNASGRPLPLGPPTRQHLLFRSLSQN
TATQGLLRGAMALAPPPRGLWQAKDISPIVKAARRDGELLLSAQLDSEERDPGVPRPSPY
APYILVYANDLAISEPNSVAVTLQRYDPFPAGDPEPRAAPNNSADPRVRRAAQATGPLQD
NELPGLDERPPRAHAQHFHKHQLWPSPFRALKPRPGRKDRRKKGQEVFMAASQVLDFDEK
TMQKARRKQWDEPRVCSRRYLKVDFADIGWNEWIISPKSFDAYYCAGACEFPMPKVDAYS
VASAGEQQQSSMAWDCEDGMGAWIVRPSNHATIQSIVRAVGIIPGIPEPCCVPDKMNSLG
VLFLDENRNVVLKVYPNMSVDTCACX
```

Figure 5. Output from GENSCAN. The exon prediction of the human DNA sequence D49493. The information used in this project from this output is the *Begin* and *End* positions of the predicted exons and the associated probability *P*.

The probability assigned to a predicted exon region is normalized between 0.2 and 0.8. A value of 0.2 is a non-predicted nucleotide, while 0.8 correspond to a curtain prediction. The normalization is explained in Figure 8.

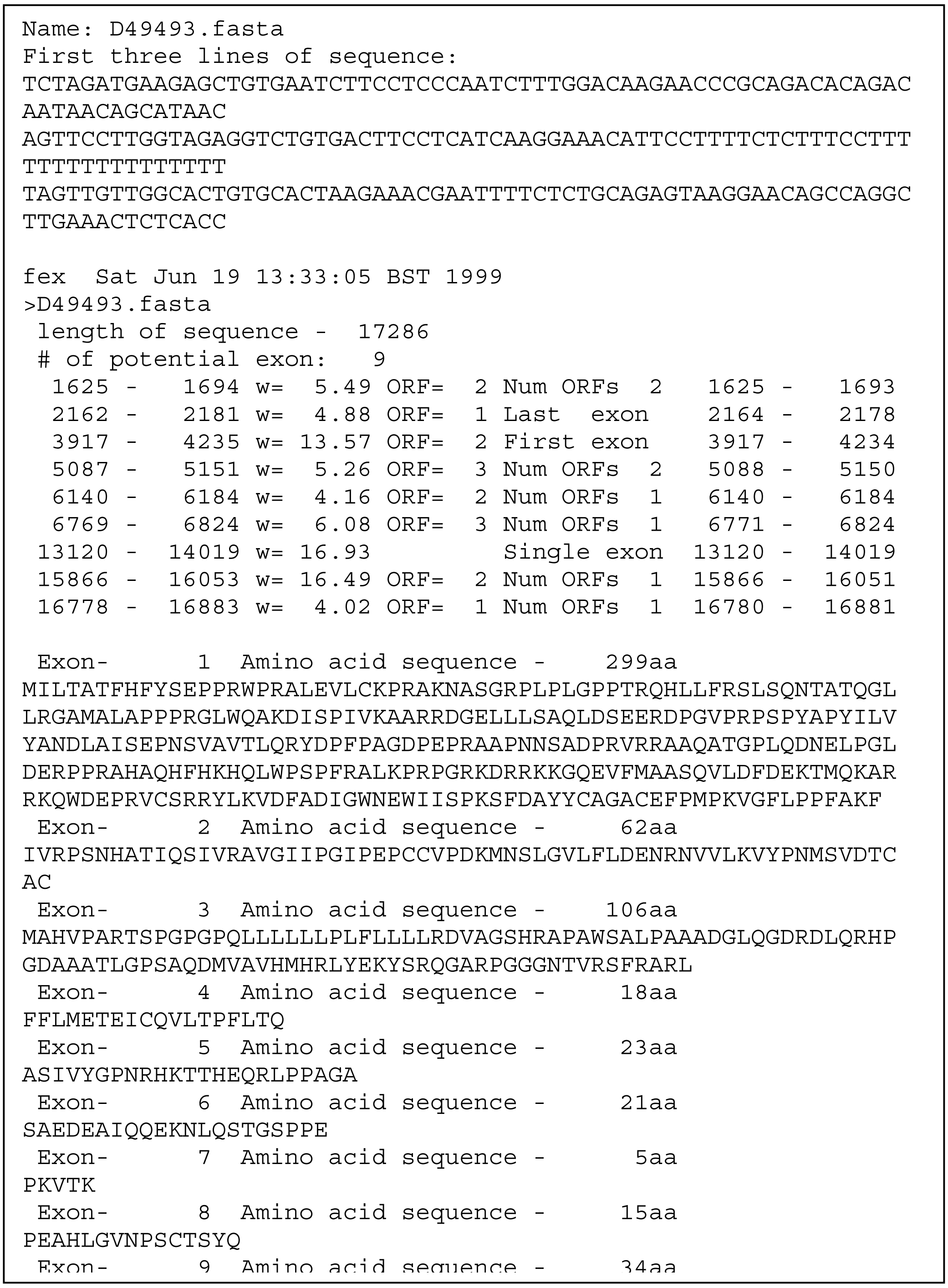
2.2.3 FEXH

FEXH was developed at the Department of Cell Biology at Baylor College of Medicine (Solovyev et. al., 1994). In this project FEXH version 2 will be used which was released in May 1994. The department has released newer and more accurate programs since then, but this version will be used to keep the similarity to the experiments done by Murakami and Takagi (1998) as high as possible. FEXH is available through WWW and as an e-mail service[5].

The algorithm FEXH used to predict exons is briefly described in two steps. Firstly, all internal exons in the sequence are predicted using a linear discriminant function that combines the characteristics that describe donor and acceptor splice sites, 5’-and 3’-intron regions and also the coding region for each open reading frame flanked by GT and AG base pairs. Then, the potential 5’-and 3’-exons are predicted by discriminant functions on the left side of the first internal exon and on the right side of the last internal exon.

In Figure 6 the FEXH exon prediction for the human DNA sequence D49493 is shown. Start and end positions of the predicted exons are given together with weights (*w*) that reflect the prediction certainty. The information about *open reading frames* (*ORF*) is not used in this project. For more details see Solovyev and Salamov (1997) and Solovyev et.al. (1994).

[5] FEXH. http://genomic.sanger.uk/gf/gf.shtml



```
Name: D49493.fasta
First three lines of sequence:
TCTAGATGAAGAGCTGTGAATCTTCCTCCCAATCTTTGGACAAGAACCCGCAGACACAGAC
AATAACAGCATAAC
AGTTCCTTGGTAGAGGTCTGTGACTTCCTCATCAAGGAAACATTCCTTTTCTCTTTCCTTT
TTTTTTTTTTTTTT
TAGTTGTTGGCACTGTGCACTAAGAAACGAATTTTCTCTGCAGAGTAAGGAACAGCCAGGC
TTGAAACTCTCACC

fex  Sat Jun 19 13:33:05 BST 1999
>D49493.fasta
 length of sequence -  17286
 # of potential exon:   9
  1625 -   1694 w=  5.49 ORF=  2 Num ORFs  2   1625 -   1693
  2162 -   2181 w=  4.88 ORF=  1 Last  exon    2164 -   2178
  3917 -   4235 w= 13.57 ORF=  2 First exon    3917 -   4234
  5087 -   5151 w=  5.26 ORF=  3 Num ORFs  2   5088 -   5150
  6140 -   6184 w=  4.16 ORF=  2 Num ORFs  1   6140 -   6184
  6769 -   6824 w=  6.08 ORF=  3 Num ORFs  1   6771 -   6824
 13120 -  14019 w= 16.93         Single exon  13120 -  14019
 15866 -  16053 w= 16.49 ORF=  2 Num ORFs  1  15866 -  16051
 16778 -  16883 w=  4.02 ORF=  1 Num ORFs  1  16780 -  16881

 Exon-      1  Amino acid sequence -    299aa
MILTATFHFYSEPPRWPRALEVLCKPRAKNASGRPLPLGPPTRQHLLFRSLSQNTATQGL
LRGAMALAPPPRGLWQAKDISPIVKAARRDGELLLSAQLDSEERDPGVPRPSPYAPYILV
YANDLAISEPNSVAVTLQRYDPFPAGDPEPRAAPNNSADPRVRRAAQATGPLQDNELPGL
DERPPRAHAQHFHKHQLWPSPFRALKPRPGRKDRRKKGQEVFMAASQVLDFDEKTMQKAR
RKQWDEPRVCSRRYLKVDFADIGWNEWIISPKSFDAYYCAGACEFPMPKVGFLPPFAKF
 Exon-      2  Amino acid sequence -     62aa
IVRPSNHATIQSIVRAVGIIPGIPEPCCVPDKMNSLGVLFLDENRNVVLKVYPNMSVDTC
AC
 Exon-      3  Amino acid sequence -    106aa
MAHVPARTSPGPGPQLLLLLLPLFLLLLRDVAGSHRAPAWSALPAAADGLQGDRDLQRHP
GDAAATLGPSAQDMVAVHMHRLYEKYSRQGARPGGGNTVRSFRARL
 Exon-      4  Amino acid sequence -     18aa
FFLMETEICQVLTPFLTQ
 Exon-      5  Amino acid sequence -     23aa
ASIVYGPNRHKTTHEQRLPPAGA
 Exon-      6  Amino acid sequence -     21aa
SAEDEAIQQEKNLQSTGSPPE
 Exon-      7  Amino acid sequence -      5aa
PKVTK
 Exon-      8  Amino acid sequence -     15aa
PEAHLGVNPSCTSYQ
 Exon-      9  Amino acid sequence -     34aa
```

Figure 6. FEXH prediction of the human DNA sequence D49493. The predicted exons are given start and end positions in the sequence and an associated weight which reflects the quality of the prediction.

In this project the by FEXH predicted exon-regions are coded using the weight associated with the predictions. The weight is a linear discriminant value that might be any, but in practice the weights are around 0. –10 and +100 are probably the limits. (Solovyev et.al. 1994) . To normalize the weights between 0.2 and 0.8 the standard deviation was computed on the whole data set. The format of the predictions after normalization is shown in Figure 8.

2.3 GenBank

GenBank is a database containing nucleic acid sequences gathered from patents and journal literature and direct author submissions. The producer of the database is the National Center for Biotechnology Information, USA [6]. GenBank is updated daily and reloaded weakly.

The nucleic acid sequences contained in GenBank have related descriptive information such as source organism, description, sequence length, and references.

For all the sequence in the data set the information about exons in GenBank will be used as the actual exons. In the data files the CDS will be used as in Murakami and Takagi’s (1998) experiments.

In Figure 7 the GenBank entry for the human DNA sequence D49493 is shown. It is only a small part of the file. The complete file can be seen in appendix A. The CDS-regions marked in the file can be compared to the predictions made by the separate programs on the same sequence (see Chapter 2.2). This exemplifies the problem of combining the different predictions to gain the best overall prediction.

```
FEATURES             Location/Qualifiers
     source          1..17286
                     /organism="Homo sapiens"
                     /db_xref="taxon:9606"
     exon            <3490..4235
                     /number=1
     5'UTR           <3490..3916
     CDS             join(3917..4235,13061..13986,15866..16057)
                     /codon_start=1
                     /product="human bone morphogenetic
protein-3b"
                     /protein_id="BAA08453.1"
                     /db_xref="PID:d1009064"
                     /db_xref="PID:g699606"
                     /db_xref="GI:699606"

/translation="MAHVPARTSPGPGPQLLLLLLPLFLLLLRDVAGSHRAPAWSALP

AAADGLQGDRDLQRHPGDAAATLGPSAQDMVAVHMHRLYEKYSRQGARPGGGNTVRSF

RARLEVVDQKAVYFFNLTSMQDSEMILTATFHFYSEPPRWPRALEVLCKPRAKNASGR

PLPLGPPTRQHLLFRSLSQNTATQGLLRGAMALAPPPRGLWQAKDISPIVKAARRDGE

LLLSAQLDSEERDPGVPRPSPYAPYILVYANDLAISEPNSVAVTLQRYDPFPAGDPEP

RAAPNNSADPRVRRAAQATGPLQDNELPGLDERPPRAHAQHFHKHQLWPSPFRALKPR

PGRKDRRKKGQEVFMAASQVLDFDEKTMQKARRKQWDEPRVCSRRYLKVDFADIGWNE

WIISPKSFDAYYCAGACEFPMPKIVRPSNHATIQSIVRAVGIIPGIPEPCCVPDKMNS
                     LGVLFLDENRNVVLKVYPNMSVDTCACR"
     intron          4236..13060
                     /number=1
     exon            13061..13986
                     /number=2
     intron          13987..15865
                     /number=2
     exon            15866..>16827
                     /number=3
     3'UTR           16058..>16827
     polyA_signal    16810..16815
```

Figure 7. A part of the GenBank entry for the human DNA sequence D49493. The regions marked by *CDS* are used as actual exons.

[6] National Center for Biotechnology Information, 8600 Rockville Pike, Bethesda, MD 20892, USA.

The predicted exon regions are assigned the score 0.8, while all other nucleotides have the value 0.2. This is shown in Figure 8.

2.4 Combination of results from gene-finding programs

Many researchers have discussed the unsatisfying results of most gene-finding programs available today (e.g. Rogozin et. al., 1998). Burset and Guigó (1996) state on page 355 in their article:

Thus, although the current generation of programs may still be of great use in pinpointing some of the regions containing exons in large DNA genomic sequences, the programs are unlikely to be able, in most cases, to elucidate their genomic structure completely.

Burset and Guigó (1996) discussed the possible advantages of combining gene-finding program results. The initial experiment results they presented suggested that a combination of gene-finding programs might be useful if an appropriate combination method is found.

In a review in *Trend in Genetics* in August 1996 James Fickett discussed the state of the art of gene-finding programs and how working principles and strengths of different programs (methods or algorithms) could be combined to gain the optimum gene identification accuracy. Fickett suggested development of a framework for program integration allowing users to integrate any set of programs.

One of the first attempts to combine results from several gene-finding programs was made by Murakami and Takagi (1998). They tested four different gene-finding programs; FEXH, GeneParser3, GENSCAN, and GRAILII. The results from the programs were combined using the five methods AND, OR, HIGHEST, RULE and BOUNDARY (see Chapter 1 for description of the methods) and the results were evaluated.

The human DNA sequences in their data set were collected from GenBank release 100 (April 1997), using a number of selection criteria described in Chapter 3.2. In the article Murakami and Takagi (1998) criticize earlier gene-finding program evaluations which were done using data sets that were not sufficiently documented. The training set and test set used are not always clearly defined e.g. there might be sequences in test sets that are very similar to sequences used in training set. The uncareful selection of sequences in the dataset might be the explanation why tests done by program developers show clearly better results than the test done by Murakami and Takagi (1998).

The results presented in the article (Murakami and Takagi, 1998) showed that by two of the methods the approximate correlation (see Chapter 3.3 for definition) was improved by 3-5 % in comparison with the best single gene-finding program (GENSCAN) tested. The methods OR, HIGHEST and BOUNDARY show increasing accuracy as the number of gene- finding programs increase. The accuracy decreases as the number of programs combined increase with the AND method. The tests show that one program, GENSCAN, has an accuracy that is difficult to increase by combining the program with any of the other three programs (FEXH, GRAIL and GeneParser3).

The experiments results presented in the article by Murakami and Takagi (1998) inspired the work in this project. As stated in the thesis statement in Chapter 1 three new methods for combination of gene-finding programs are evaluated in this project. Two of the methods are based on ANNs and they are expected to perform better than the logical methods AND, OR, and MAJ.

2.5 Chapter summary

In this Chapter the background for the project is outlined briefly. The necessary biological background is described to introduce the reader to the problem domain. It is important to understand the distinction between DNA, genes, exons, and introns. There are also facts about the DNA that will give the reader an understanding of the complexity in gene-searching.

Many different computer based techniques have been tried for the gene-searching problem. In this Chapter some techniques are presented and there are references to a number of researchers using different techniques. The three well-known gene-finding programs used in this project are presented (i.e. GRAILII, FEXH, and GENSCAN). GenBank, the enormous database of DNA sequences is briefly introduced to the reader. It is the exons in GenBank files of the sequences in the data set that will be used as the actual exons.

Researchers that have discussed the advantages of combining results from gene-finding programs are presented. The key background for this project is the article published by

Murakami and Takagi (1998). Results of the evaluation of methods for combining gene-finding program results are discussed.

The chapter in its whole illustrates the motivation for the aim and hypothesis addressed in this project.

3 Experiments

The aim of this project is to compare different methods for combining results from gene-finding programs. The experiments that are performed within this project are presented in this Chapter.

In addition to the logical combination methods inspired by Murakami and Takagi (1998) this project includes the use of ANNs to combine the results of the gene-predicting programs. Two types of ANNs will be used; basic feed-forward neural networks and feed-forward neural networks using sliding windows.

3.1 Overview of experiments

The process through which the hypothesis will be fulfilled or falsified is show in Figures 8 and 9.

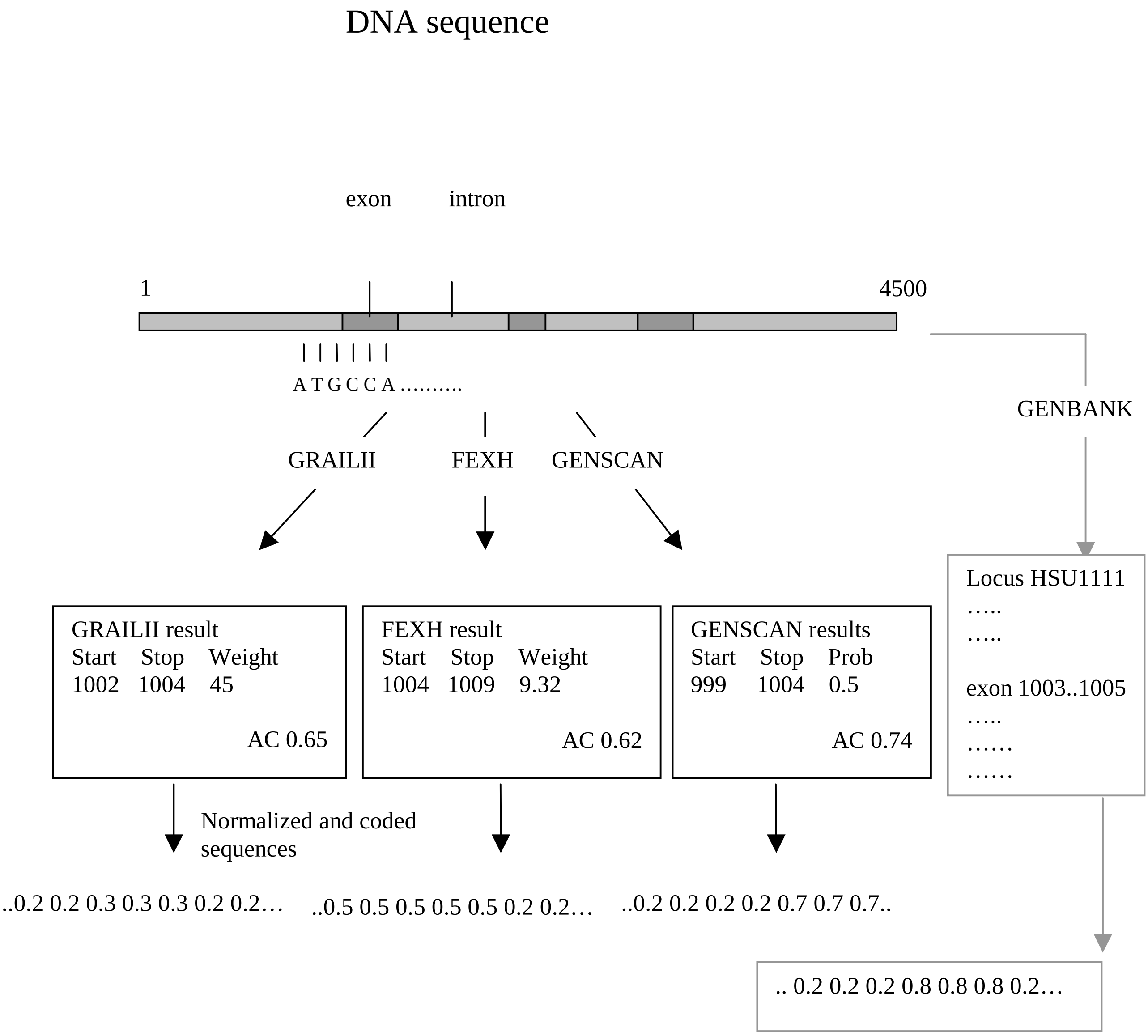


Figure 8. The first steps in the process of this project. For explanation see the text.

All sequences in the data set are analyzed by all the gene-finding programs used in this project (i.e. GRAILII, FEXH and GENSCAN). In Figure 8 there are examples of the output from the programs. All three programs give the results in this form (See Chapter 2.2). The start and the stop positions for a predicted exon are given with a weight, score, or probability to illustrate the probability of the prediction being correct. The predicted sequence structures are built using these outputs normalized between 0.2 and 0.8 (this is just to give the ANN a more manageable scale of inputs). To the right in the figure the correct sequence structure is fetched from GenBank. The actual sequence structure is encoded in the same way as the predicted structures. The minimum value (i.e. 0.2) is used for all non-coding nucleotides and the maximum value (i.e. 0.8) is used for positions with true coding nucleotides. The rest of the process is illustrated in Figure 9.

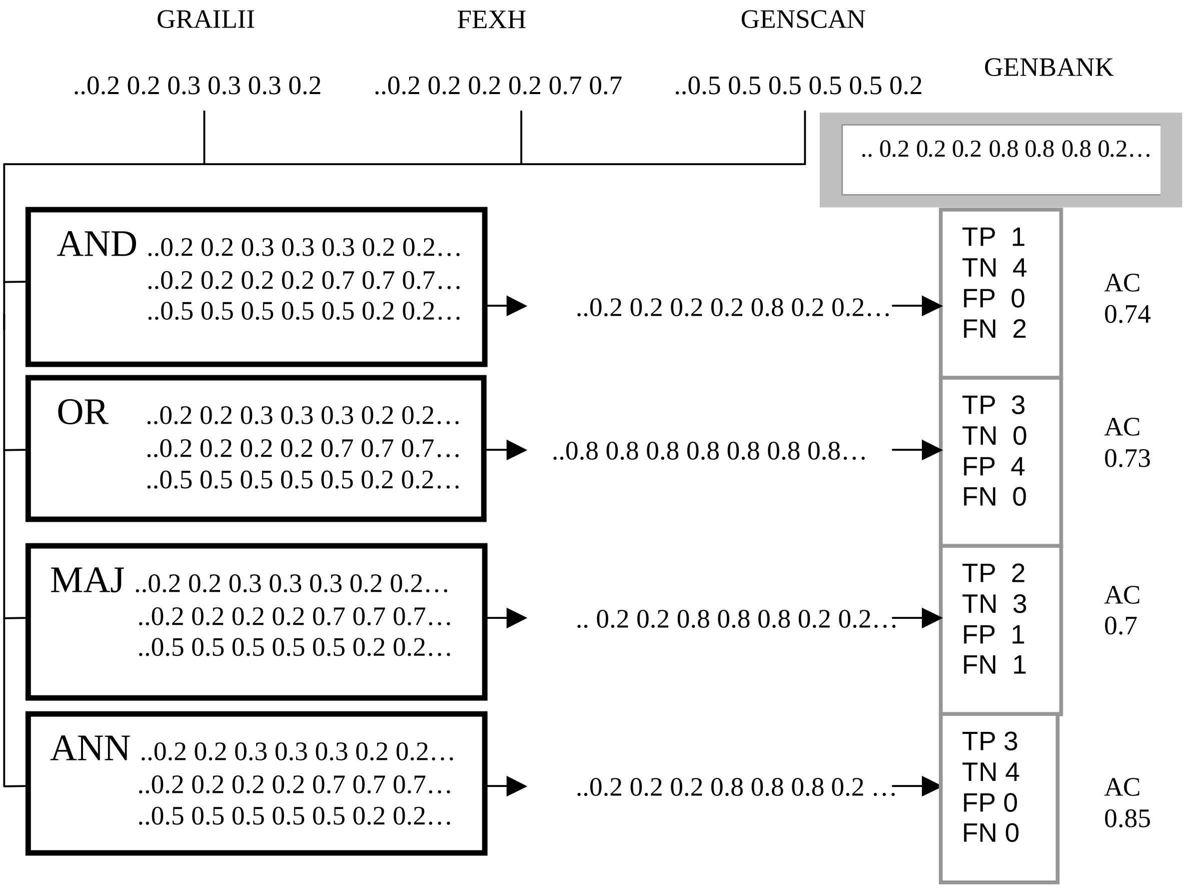


Figure 9. The steps of the project process where the results from different exon-predicting programs are combined using different methods. This process is explained in detail in the text.

The sequences are combined using the four different methods (i.e. AND, OR, MAJ, ANN). With the AND method all program predictions need to be positive (i.e. greater than 0.2 that represent negative, non-coding regions) at a certain position in the sequence for the overall prediction to be positive. With the OR method only one of the predictions at a certain position needs to be positive for the resulting prediction at that position to be positive. The MAJ method needs at least two programs to give a positive prediction for the overall prediction to be positive. The ANN might adopt a strategy in between the other methods.

The resulting predictions after combining the individual program predictions are compared to the sequence structure found in GenBank. To calculate the accuracy on the exon level every position of the predicted sequence structure is compared to the true sequence structure. The number of *True Positives* (*TP*) is calculated which shows the number of correctly predicted positives. The number of correctly predicted negatives, called *True Negatives* (*TN*), is calculated. The *False Positives* (*FP*) are the incorrectly predicted positive and the *False Negatives* (*FN*) are the incorrectly predicted negative. The variables are used to calculate the *Approximate Correlation* (*AC*) (i.e. the association between prediction and reality) and other measures used to illustrate the accuracy of the prediction. (See Chapter 3.3).

In the Figure 8 example AC values are given for the programs when used separately on the data set in this project. Example values of AC after using the combination methods are shown in Figure 9. Murakami and Takagi (1998) showed that in general the AC was improved by a combination of programs. The AND and OR methods used here are inspired by Murakami and Takagi's (1998) experiments.

3.2 Data set

The data set Murakami and Takagi (1998) chose for their project will be used in this project. The criteria they used when choosing DNA sequences from GeneBank are well documented (see Figure 10) and using the same data set will contribute to a possible comparison between results in the projects, see discussion in Chapter 9.

From GenBank Release 100 (April 1997) (Murakami et. al., 1998) collected loci that meet these criterias:

- Human DNA sequences with at least one ‘CDS’
- ‘SOURCE’ is Homo sapiens
- standard splice site conservative dinucleotides (i. e. GT-AC)
- confirmed experimentally to be transcribed
- immunoglobin genes discarded due to the complexity of their structure
- registrated after June 1996
- do not code for proteins that are homologous to already known proteins

Figure 10. The criteria for loci selection from GenBank used by (Murakami et al, 1998).

The process of collecting sequences from GenBank resulted in a data set of 219 loci[7] that were randomly divided in two parts; training set and test set. The training set consists of two thirds of the data set (146 loci) and the test set consists of the remaining third (73 loci). These data sets are available upon request from Murakami and Takagi (1998). In appendix A there is a list of all 219 loci with name and sequence length.

The lengths of the sequences are spread over a spectrum as the graph shows in Figure 11.

[7] Loci. Pl. For locus. Locus is an accession number of ensured unikeness in GenBank.

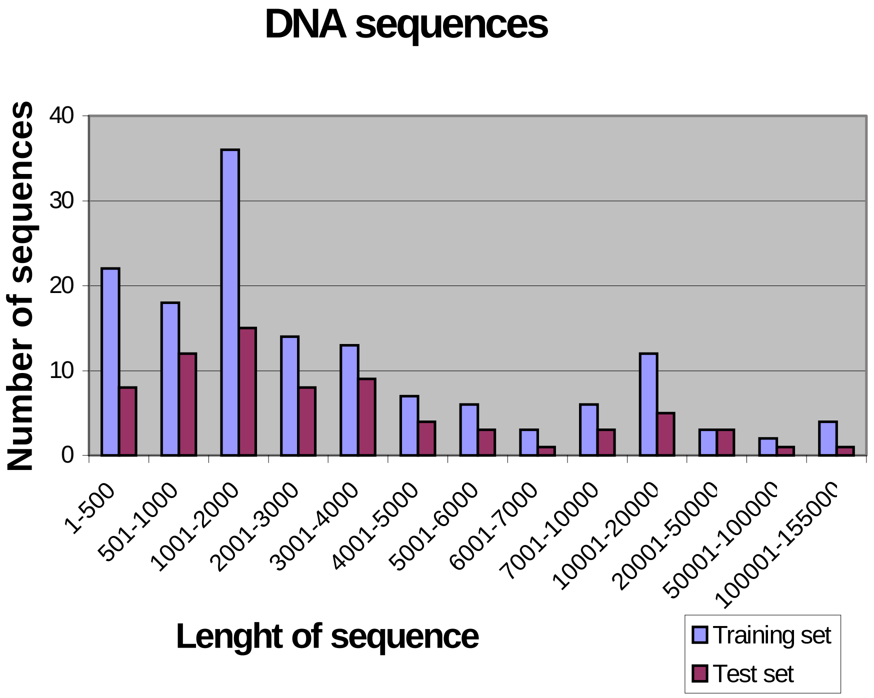


Figure 11. The lengths of the sequences in the training set and test set. Note that the ranges of sequence lengths are different.

3.3 Performance measures

The accuracy of the predictions of gene-finding programs is often measured at three different levels; nucleotide level, exon level, and protein product level (Burset and Guigó, 1996). At the coding nucleotide level the prediction accuracy is measured by comparing the prediction (coding or non-coding nucleotide) with the actual nucleotide value (coding or non-coding) for each nucleotide separately along the sequence. At the exon level the accuracy of the prediction is measured by comparing the predicted exons with the actual exons along the sequence. An exon that is correctly predicted is generally defined as an exact match with the true exon (i.e. has to include both the start and the stop codon). The accuracy of predictions measured at the protein product level is performed by comparing the protein product encoded by the predicted gene with the protein product encoded by the actual gene. This approach gives a measure of how well the predicted exons assemble into the correct protein product. High similarity between such a predicted protein product and a known protein sequence may improve the prediction.

In this project only prediction accuracy at the nucleotide and exon levels is considered. There are four well recognized measures of performance used when evaluating gene-predicting programs; *Sensitivity, Specificity, Correlation Coefficient and Approximate Correlation.* The measures are used in other areas as well (e.g. protein folding prediction (Sternberg, 1997)). The measures are explained in more detail in the following subchapters. These measures are commonly used and they are also used in this project for comparison reason. There are some problems associated with the measures however, and these are discussed in the following subchapters.

3.3.1 Nucleotide level

A common approach to measure prediction accuracy is at the nucleotide level (see e.g. Murakami and Takagi, 1998, Burset and Guigó, 1996). The actual nucleotide value as well as the predicted nucleotide value can be either coding or non-coding (positive or negative) and associations between the prediction value and the actual value have been widely used as measures of prediction accuracy. The values (coding or not) of the real DNA sequence compared to the values in the prediction are shown in Figure 12. The top square in Figure 12 represents the nucleotides of the DNA sequence to be analyzed. These are divided in two groups; actual positives and actual negatives. The square in the middle represents possible outputs from a gene-finding program. The gene-finding program does not give 100% correct predictions. Most of the actual positive nucleotides are predicted as being positives and most of the actual negatives are predicted negative. There are some actual negative nucleotides that are predicted as positives, as well as there are some actual positives predicted as negatives. When comparing the predicted values of the nucleotides with the actual values, the result is a number of correct positive predictions (True Positives or TP), and a number of correct negative predictions (True Negatives or TN). The comparison between the prediction and the actual sequence also results in a number of incorrectly positive predictions (False Positives or FP), and a number of incorrectly negative predictions (False Negatives or FN). The results of the comparison in this example are shown in the bottom square in Figure 12.

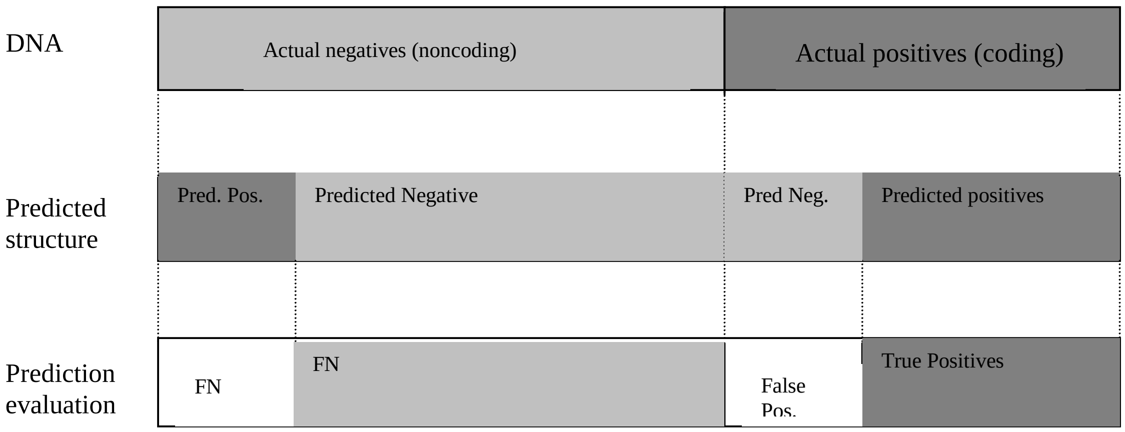


Figure 12. The variables used when evaluating a predicted DNA sequence structure. The top sequence shows the actual DNA sequence structure. For simplicity this sequence has only one non-coding region (intron) followed by one coding region (exon). The middle sequence illustrates the prediction of the DNA sequence structure. The dark regions are regions of predicted coding bases and the light regions are regions of predicted non-coding bases. The bottom sequence shows the result of the comparison of the predicted DNA sequence structure with the actual sequence structure. For every nucleotide base, the prediction is compared to the actual nucleotide base at the corresponding position in the actual sequence. For the whole sequence the total number of correctly predicted positives (i.e. True Positives), the correctly predicted negatives (i.e. True Negatives), the incorrectly predicted positives (i.e. False Positives), and the incorrectly predicted negatives (i.e. False Negatives), are counted.

The associations between the four variables (TP, TN, FP, FN) are often derived by the two measures; *Sensitivity* (*Sn*) and *Specificity* (*Sp*), which are defined in definition 1.

Sn is the proportion of predicted positive nucleotides actually being positives, while *Sp* is the proportion of actual positive nucleotides being predicted as positives.

$$Sn=\frac{TP}{(TP+FN)} \qquad Sp=\frac{TP}{(TP+FP)}$$

Definition 1. The definitions of sensitivity (*Sn*) and Specificity (*Sp*). Sensitivity is the proportion of coding nucleotides correctly predicted as coding. The specificity is the proportion of non-coding nucleotides correctly predicted as non-coding.

It is obvious that a prediction of a gene-predicting program can result in high sensitivity but very low specificity (e.g. when the program predicts all nucleotides to be positive when they are actually not) as well as high specificity together with very low sensitivity (e.g. the program predicts all nucleotides as negatives when there are a number of actual positives). The sensitivity and specificity measures do not reflect the size of the data set. The measures used separately do not give a good picture of the prediction accuracy.

Correlation Coefficient (CC) is a measure widely used for evaluation of the gene-prediction accuracy. CC ranges fro –1 to 1 and reflects the association between the prediction and the reality. Definition 2 gives the formal definition of CC. CC is an appropriate measure of the overall prediction accuracy since it depends on both the sensitivity and specificity in predicting positive nucleotides, as well as sensitivity and specificity in predicting negative nucleotides.

CC has one drawback. In situations were the prediction or the reality does not contain either any positives or any negatives. CC is not defined for situations when TP+FN, TP+FP, TN+FP, or TN+FN is zero.

$$CC=\frac{((TP\times TN)-(FN\times FP))}{\sqrt{((TP+FN)\times(TP+FP)\times(TN+FP)\times(TN+FN))}}$$

Definition 2. Correlation Coefficient (CC).

Approximate correlation (AC) is also a widely used accuracy measure. It is defined in definition 3. AC ranges from –1 to 1 and appears to be an approximation of CC. It can therefore be used as an alternative to CC. (Burset and Guigo, 1996).

$$AC=\frac{1}{2}\left[\frac{TP}{AP}+\frac{TP}{PP}+\frac{TN}{AN}+\frac{TN}{PN}\right]-1$$

Definition 3. Approximate Correlation (AC). AC shows the association between the prediction and reality.

3.3.2 Exon level

When measuring accuracy at the exon level the comparison between the prediction and the reality with regard to the complete exons (Figure 13). The top square in Figure 13 represents the actual structure of the DNA sequence to be analyzed. In the middle square the results from a gene-finding program are shown. The predicted exons do not match exactly with the actual exons in the DNA sequence. To be considered in the accuracy measures the predicted exon has to match exactly with the actual exon. The bottom square shows the correct predicted exon along with a missing exon (M) and a wrong exon. A missing exon is defined as an actual exon not being overlapped with a predicted exon, while a wrong exon is a predicted exon that does not actually exist.

At the exon level the definition for *sensitivity* (definition 4) is interpreted as the proportion of actual exons being predicted. *Specificity* (definition 4) is the proportion of predicted exons that are actual exons.

Burset and Guigo (1996) used two additional measures of sensitivity and specificity to compensate for the stringent definition they used for correct exons. *Missing Exon* (ME) is a measure of the proportion of actual exons with no overlap to the predicted exons. *Wrong Exon* is the proportion of predicted exons with no overlap to any actual exon. The definitions for these measures are shown in definition 5.

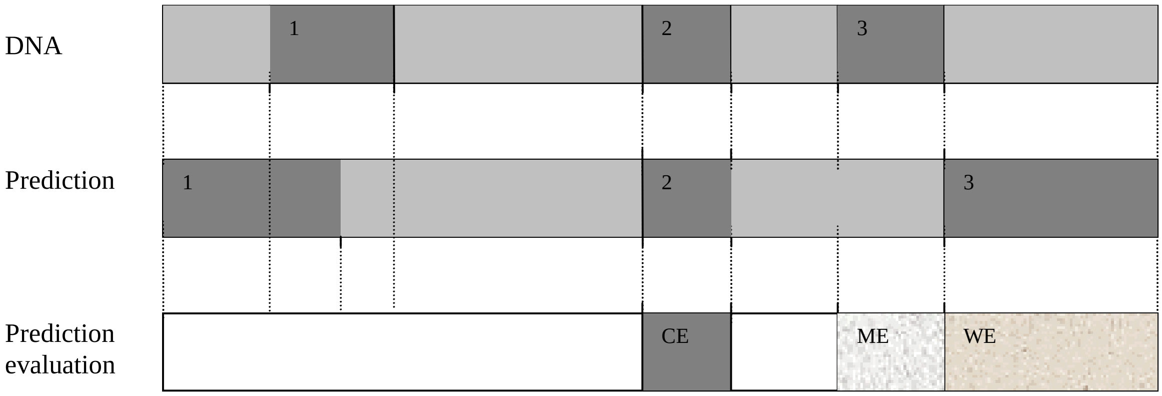


Figure 13. Variables used for accuracy measures at the exon level. In the top sequence the actual DNA sequence structure is illustrated. The dark gray regions represent the actual exons and the light gray regions represent the actual introns. The middle sequence shows a possible prediction of the sequence structure (or a resulting prediction after combining the results of different programs). The bottom sequence shows the result of the evaluation of the exon prediction. There is one exon predicted with an overlap to an actual exon. The letter symbol *CE* (i.e. correct exon) represents the exon that has been predicted exactly. The variables that will be used for accuracy measures at the exon level are the *missing exons* (i.e. *ME*) and the *wrong exons* (i.e. *WE*). A *missing exon* is an actual exon that has no overlap in any predicted exon. A *wrong exon* is a predicted exon that has no overlap in any actual exon.

$$Sn = \frac{\text{Number of Correct Exons}}{\text{Number of Actual Exons}}$$

$$Sp = \frac{\text{Number of Correct Exons}}{\text{Number of Predicted Exons}}$$

Definition 4. *Sensitivity* (*Sn*) and *Specificity* (Sp) at the exon level.

$$ME = \frac{\text{Number of Missing Exons}}{\text{Number of Actual Exons}}$$

$$WE = \frac{\text{Number of Wrong Exons}}{\text{Number of Predicted Exons}}$$

Definition 5. *Missing exons* (*ME*) and *wrong exons* (*WE*) at the exon level

In this project the accuracy of the predictions is measured using the measures as defined above. The results from the different gene-finding programs are analyzed separately as well as the combinations of the program predictions. A number of additional measures are found in the literature. See Burset and Guigo (1996) for a deeper analysis of the different measurements.

3.4 AND

One of the simplest ways of combining results from any programs like the ones used here is to use an AND operator. At the nucleotide level this means that an overall positively predicted nucleotide is only considered if all participating programs predict the nucleotide as positive. At the exon level a whole exon has to be predicted by all programs to result in an overall positive prediction of the exon. How this combination method works at the different levels is shown in Figure 14.

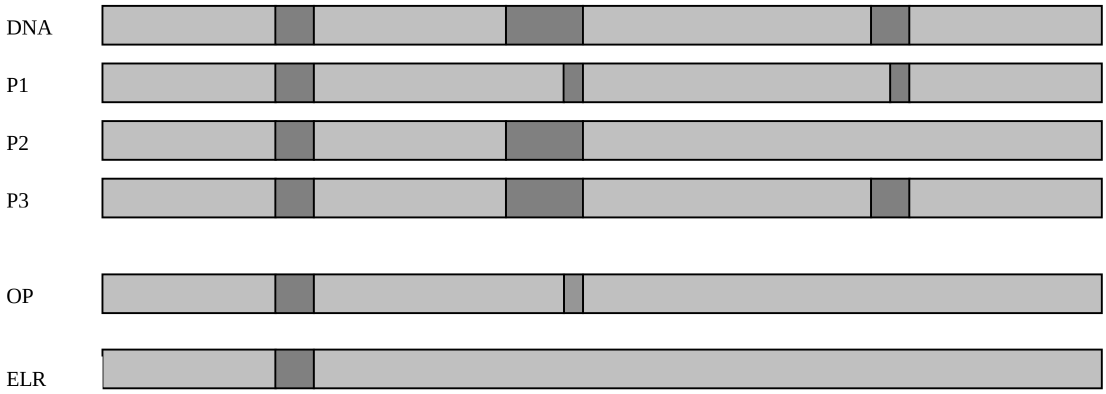


Figure 14. The AND method for combination of gene-predictions. *DNA* is the actual sequence being analyzed. *P1*, *P2* and *P3* show possible predictions by the different programs. The dark regions represent exons and the light gray regions are introns. . *OP* is the resulting overall prediction and the result at the nucleotide level. *ELR* is the result after evaluating the overall prediction at the exon level.

The AND method for combination of results from gene-finding programs was used by Murakami and Takagi (1998). The results obtained in this project are compared to the results presented in their article. There is a difference between how Murakami and Takagi (1998) used the output from the gene-finding programs and how the scores are used here. They transformed the scores of the predicted exons given by the different programs to probabilistic ones. The reason they give for the transformation of scores is that they wanted to be able to compare the quality of the predictions from different programs. The transformation was done by examining the relationship between the scores and the accuracy of the programs (Murakami and Takagi, 1998). This demands some consideration, for instance when adding a new program to the AND method. Most gene-finding programs have their own way of computing the score, weight or probability of their predictions which might result in difficulty when transforming the predictions to comparable probabilities.

In this project the weights, scores and probabilities given by the different programs with the predictions will be normalized to facilitate use of this data as input to the ANN.

The results from the AND method will also be compared to the results obtained by the different gene-finding programs separately.

The AND method is expected to result in low sensitivity and high specificity, as understandable and as shown by Murakami and Takagi (1998). The sensitivity will generally stay the same or most likely decrease as the number of combined programs increases. Because the method needs a positive prediction by all the programs to give an overall positive prediction, all participating programs have to recognize the same particularities that unveil exons in the sequence. Since each program considers the features of the sequences differently the predictions are often not identical and one program might grasp some features of a gene that others do not (Murakami and Takagi, 1998). The possibility of gaining the advantages from different approaches was the motivation for the idea to combine gene-finding programs based on different techniques. The AND method does not take this program-unique gene-feature recognition into account.

The three possible combinations of two programs among the three programs will be analyzed as well as the combination of all three programs.

3.5 OR

In Figure 15 the method is shown for the nucleotide and exon levels. For an overall positive prediction of a nucleotide there has to be at least one program predicting the nucleotide as positive. In the same way an exon has to be predicted by at least one program to be considered in the overall positive prediction.

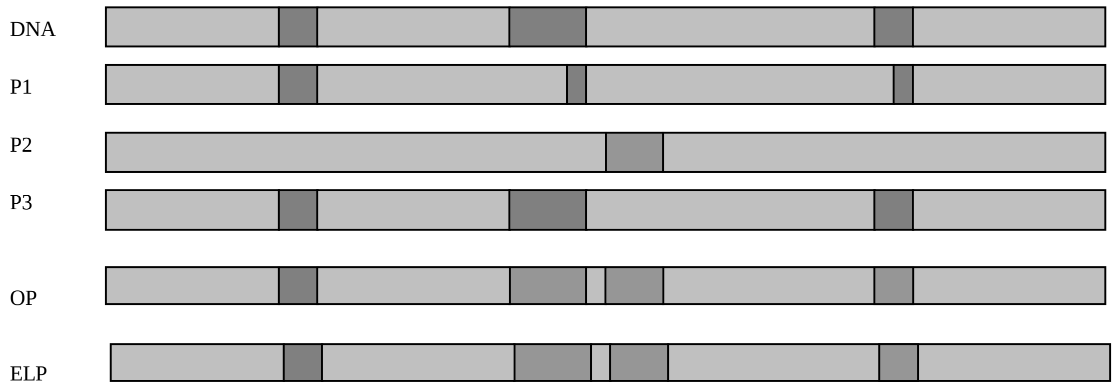


Figure 15. The OR method for combination of predictions. *DNA* is the actual DNA sequence. *P1*, *P2*, and *P3* are possible predicted structures of the DNA sequence performed by programs. Dark gray regions represent exons, while light gray regions represent introns. *OP* is the overall prediction after combining P1, P2, and P3 using the OR method and it is also the resulting prediction at the nucleotide level. *ELP* is the resulting exons at the exon level.

This combination method was also used by Murakami and Takagi (1998). The results obtained by the experiments in this project will be compared to the results presented in the article (Murakami and Takagi, 1998). The results obtained by the different programs when used separately will also be compared to Murakami and Takagi's (1998) results. There will also be a comparison between the different methods for combination of predictions included in this project.

The OR method is expected to consider all exons/nucleotides predicted by the programs. The method will result in low specificity and high sensitivity. A nucleotide/exon predicted as negative is more likely to be actually negative with this method than with AND. The probability that a positive prediction matches an actual positive nucleotide/exon is lower with this method than with AND. The OR method considers all program-unique gene-feature recognitions there are among the combined programs, but it also includes all the programs-unique weaknesses too (e.g. incorrect predictions in situations recognized as coding by a program). The sensitivity is likely to increase as the number of programs increases, while the specificity will decrease as the number of combined programs increases.

The test setups are the same as for the AND method. All possible combinations of two programs among the three will be tested as well as the combination of all three programs.

3.6 Majority

The AND method only considers the nucleotides/exons which are predicted similarly by all the programs, while the OR method considers all the predictions done by the programs. MAJORITY (MAJ) is a method which considers the predictions done similarly by the majority of the programs (i.e. by at least two programs).

The expected results for this method is a sensitivity and a specificity between those of AND and OR. MAJ might result in a higher overall accuracy, AC, when compared to the AND and OR methods of combining three programs.

The MAJ method is tested on only one combination of all three programs (i.e. GRAILII, GENSCAN, and FEXH). Since the MAJ method implies that more than 50% of the programs predict an exon the method is only applicable on an odd number of programs.

3.7 ANN

An artificial neural network (ANN) is a way of learning a function using a network of simple arithmetic computing elements and example data. For a full explanation of the details of ANNs see for instance Russell and Norvig (1995).

The results from the gene-finding programs are predictions of where in the DNA sequence there are coding nucleotides (i.e. possible exons). These exon predictions are assigned scores, weights or probabilities (here referred to as scores) by the different programs reflecting the certainty of the prediction being correct. These scores will be normalized on a scale between 0.2 and 0.8 to encourage sensitivity in the ANN.

The output from the ANN is an overall prediction of every nucleotide along the sequence being coded or not.

Two variants of presenting input data to the ANN are tested. Both variants are simple feed-forward networks. The first ANN (ANN type 1) is given as input one position of a time along the sequence. ANN type 2 is given as input the nucleotide position at interest as well as the close neighborhood on both sides in the sequence (ANN type 2).

The network configurations are shown in Table 1 in Chapter 3.7.3.

3.7.1 ANN type 1

The topology of the ANN type 1 is shown in Figure 16. The prediction results from the different programs are coded as described in Chapter 3.1 and fed into the network nucleotide by nucleotide. There is one input node for each of the participating programs. The pedictions of the nucleotide sequences are fead into the network one nucleotide at the time. The output from the network is the overall prediction of the nucleotides, one at the time. A threshold is used to divide the output into the two classed *predicted positive* or *predicted negative*. Initially the threshold is set to 0.5. The idea is that the ANN should learn which combinations of scores from the different programs that reflect actual coding nucleotides.

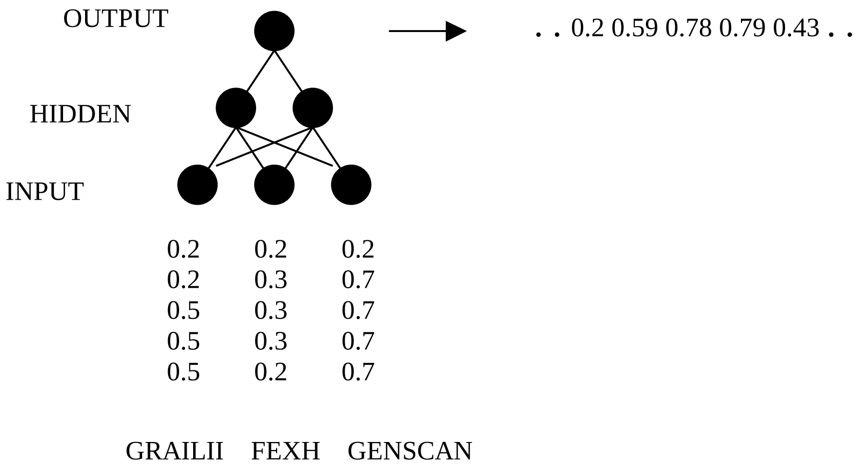


Figure 16. The topology of the ANN type 1. In the figure the ANN combines results from three programs. A threshold is set to divide the output from the ANN (i.e. predicted coding or predicted non-coding) and decide the overall prediction for every nucleotide along the sequence.

3.7.2 ANN type 2

The inputs are presented to the ANN using a sliding window. The principle of this is shown in Figure 17. It is not only the nucleotide of current interest (i.e. the nucleotide to be predicted) that is presented to the ANN. Four nucleotide predictions before and four nucleotide predictions after the nucleotide of interest are also included in the input. Three windows of nine nucleotides in the sequence (i.e. one window from each program) are fed into the ANN. The ANN will combine the predictions in the windows and give a resulting overall prediction for the nucleotide in the middle. Using this technique the ANN will have more information about the environment (i.e. the neighbor nucleotides).

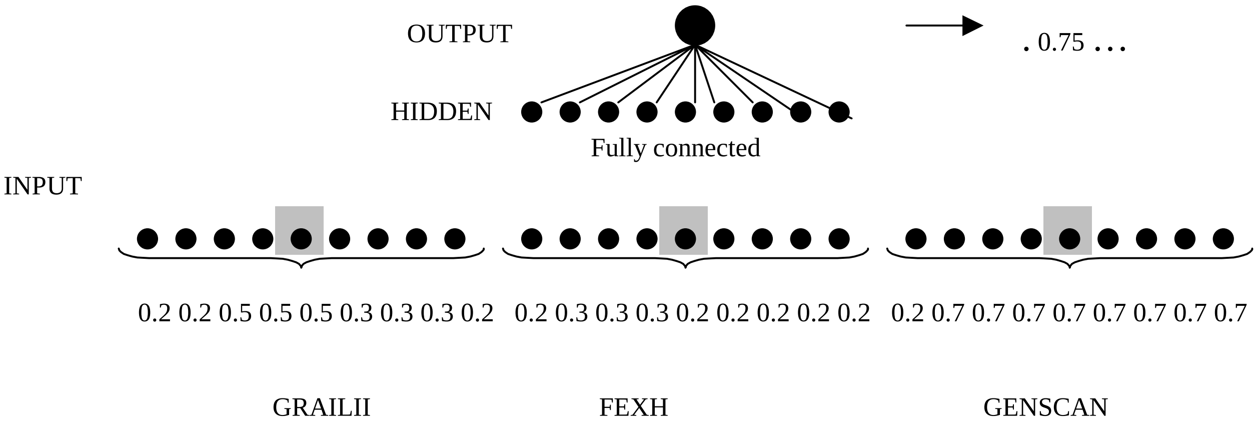


Figure 17. The ANN type 2 network in the figure is fully connected. The ANN combines the results from the gene-finding programs. Predictions at every position (i.e. nucleotide) in the sequence are presented to the ANN together with four nucleotides on each side. The output from the ANN is the overall prediction of the nucleotide in the middle of the windows being coded or not (i.e. GRAILII (0.5), FEXH (0.2), and GENSCAN (0.7)). The windows are sliding one position at the time for every position along the sequence making a new input example for the ANN.

3.7.3 Test setup

The test setup for the ANN type 2 is shown in Table 1. The symbol GR stands for GRAILII, FE stand for FEXH and GS is GENSCAN. The tests are run using Matlab Neural Network Toolbox and the training will continue until the minimum error is found or until 9000 epochs of training is reached. If the networks are still learning, i.e. if the output error is still decreasing, the network will be trained for an additional 9000 epochs. The networks are evaluated every 1000 epochs of training and the threshold between predicted non-coding and predicted coding nucleotides will be initially set to 0.5. The threshold can be varied to find out if this can improve the accuracy measures but this is not done within this project. Two architectures are tested with both the ANN type 1 and ANN type 2. As can be seen in Table 1 the most experimental effort is spent on the combination of three programs. One basic thought behind this project was that more programs would give a more accurate prediction and the results in this project will show if that is true. The other basic thought was that the ANN should perform as well as or better than the logical methods for combination of predictions. Because of limitations of time in this project only a small selection of ANN architectures can be used. The selected architectures are the simplest and most basic ones.

Test	ANN type	Programs	Input nodes	Hidden nodes	Output nodes
1	1	GR+FE	2	2	1
2	1	GR+GS	2	2	1
3	1	GS+FE	2	2	1
4	1	GR+GS+FE	3	2	1
5	1	GR+GS+FE	3	5	1
6	2	GR+GS+FE	27	9	1
7	2	GR+GS+FE	15	10	1

Table 1. The test setups for ANN type 1 and type 2. *Test1*, *2*, *3*, and *4* are all ANN type 1. *Test 5* is a ANN type 2. The programs that are combined in the different experiments are shown in column two. The number of input nodes and hidden nodes are shown in columns three and four. All ANNs have only one output node, shown in column five.

3.9 Chapter summary

This Chapter presented the resources that are used in this project and the experiments that are performed in order to fulfill the objectives.

The two Figures (8 and 9) describe the process through which this project will fulfil or falsify the hypothesis.

The data set used in this project is the same as Murakami and Takagi (1998) used. When collecting the DNA sequences from GenBank Murakami and Takagi (1998) used seven criteria that are described in Figure 10. Murakami and Takagi (1998) divided the data set in a training set (146 sequences) and a test set (73 sequences) that are used in this project.

To measure the performance of a gene-finding program a number of commonly used measures are defined and described. The performance measures used to evaluate the gene-finding programs and the combination methods in this project are approximate correlation, sensitivity, and specificity at the nucleotide level. The measures *missing exons*, and *wrong exons* are used to measure the prediction quality on the exon level. The distinction between nucleotide level and exon level prediction quality is explained in the chapter.

The methods (for combining the results of the gene-finding programs) used and evaluated in this project are explained in detail in this Chapter. There are three logical methods; AND, OR, and MAJ. The AND and OR methods are inspired by the AND and OR methods used by Murakami and Takagi (1998).

With the AND method all combined programs have to predict a particular nucleotide as coding for the resulting prediction of that nucleotide to be coding. With the OR method only one programs has to predict a coding nucleotide for it to be included in the resulting prediction. The MAJ method needs at least two programs out of the three to predict a certain nucleotide as coding for the resulting prediction to be coding.

Two new methods for combining results of gene-finding programs are proposed in this project. The methods are based on ANNs. The ANN type 1 and ANN type 2 are described in this chapter. As the hypothesis states the ANN method are expected to perform better than the logical methods when combining the results of gene-finding programs and thereby the ANNs are expected to improve the approximate correlation more than the methods used by Murakami and Takagi (1998).

4 Results

The results of the experiments are presented in this chapter. The results of the tree gene-finding programs are presented. The results of the combination methods AND, OR, MAJ, ANN type 1, and ANN type 2 are presented in this chapter. In the following subchapters GRAILII is often abbreviated GR, GS is GENSCAN, and FE stands for FEXH.

The results have been divided in two (training and test) to show differences in the training and test sets and also to enable the comparison to the ANN results. The ANN results are only shown for the test set.

4.1 Results from the gene-finding programs

All sequences in the complete data set (i.e. training set + test set) were analyzed by the three gene-finding programs GRAILII, GENSCAN and FEXH. Only the forward strand results are considered here, even though GRAILII gives a reverse strand prediction as well. The accuracy of the results produced by the programs separately is presented in Table 2. The results are for both the nucleotide level and the exon level. The first column shows the program's name. The number of sequences with undefined approximate correlation (AC) is shown in the second column (*UNDEF.*). These sequences are excluded when computing the average of the performance measures. *AC* is the average approximate correlation on the nucleotide level. Sn_{nucl} is the average sensitivity at the nucleotide level. The average specificity at the nucleotide level is shown under *Sp*. Missing exons (*ME*) and wrong exons (*WE*) show the result at the exon level. *ME* shows the missing exons, while *WE* shows the wrong exons.

The first value in each column is the result on the training set of 146 sequences. The second value in the column is the result on the test set that consists of 73 sequences. The results have been divided in two to show differences in the training and test sets and also to enable the comparison to the ANN results. The ANN results are only shown for the test set.

Program	**Undef**	**AC**	**Sn**	**Sp**	**WE**	**ME**
GENSCAN	19/18	0.77/0.77	0.80/0.77	0.86/0.91	0.11/0.09	0.04/0.09
GRAILII	24/16	0.71/0.64	0.75/0.66	0.83/0.80	0.16/0.17	0.09/0.18
FEXH	19/8	0.63/0.62	0.65/0.63	0.80/0.80	0.19/0.22	0.10/0.13

Table 2. Nucleotide level results: *AC* (approximate correlation), *Sn* (sensitivity) and *Sp* (specificity) are averaged over the defined results in the training and test sets. The results show a considerable difference in *AC, Sn*, and *Sp* between the best performing program (i.e. GENSCAN) and the worst (i.e. FEXH). On the other hand, FEXH predictions do result in the highest number of undefined ACs. The results at the exon level: *WE* is the measure *Wrong Exons*, and *ME* is *Missing Exons*.

FEXH shows the lowest number of undefined sequence ACs, 11-13%, which means that it predicts exons in more sequences than the other programs. GRAILII has as much as 16-21% sequences without predicted exons. GENSCAN predictions result in 13-25% sequences with no predicted exons.

As also shown by Murakami and Takagi (1998) GENSCAN gives the highest average approximate accuracy in this set of programs. It is a considerable difference between GENSCAN and FEXH, which is the worst performing program when considering AC.

The program that gives the best sensitivity is GENSCAN (0.80/0.77). GRAILII and FEXH have comparable sensitivity although GRAIL is somewhat better. GENSCAN has also the highest specificity among the programs in the set. GRAILII and FEXH have comparable specificity.

When considering the wrong exons GENSCAN performs best followed by GRAILII and FEXH. FEXH has the highest proportion of wrong exons which was expected since it is more generous with predictions.

GENSCAN also performs best when looking at the missing exon measure. FEXH has somewhat better missing exon-value than GRAILII on the test set.

4.2 AND

The results of the AND method for combination of the gene-finding programs are shown in Table 3. The results are on both the nucleotide level and the exon level. The first column shows which programs are combined. The number of sequences with undefined approximate correlation (AC) is shown in the second column (*UNDEF.*). These sequences are excluded from the computation of average of the other measures. *AC* is the average approximate correlation on the nucleotide level. *Sn* is the average sensitivity at the nucleotide level. The average specificity at the nucleotide level is shown under *Sp*. *ME* shows the missing exons, while *WE* shows the wrong exons.

The first value in each column is the result using the training set of 146 sequences. The second value in the column is the result on the test set consisting of 73 sequences. The results have been divided in two (training and test) to show differences in the training and test sets and also to enable the comparison to the ANN results. The ANN results are only shown for the test set.

AND	**Undef**	**AC**	**Sn**	**Sp**	**WE**	**ME**
GR and FE	37/23	0.68/0.66	0.62/0.59	0.91/0.90	0.07/0.11	0.14/0.24
GR and GS	29/23	0.75/0.73	0.73/0.67	0.92/0.94	0.07/0.04	0.08/0.18
GS and FE	34/22	0.72/0.73	0.67/0.65	0.92/0.97	0.05/0.03	0.11/0.15
GR and GS and FE	40/27	0.69/0.71	0.62/0.61	0.93/0.97	0.04/0.02	0.14/0.23

Table 3. The results of the AND combination method. On the nucleotide level: The number of sequences with undefined AC is shown in the second column. The average *AC* (approximate correlation*)*, *Sn* (sensitivity), and *Sp* (specificity) are given for the four possible combinations of programs. Results at the exon level: *WE* is the measure *Wrong Exons,* and *ME* is *Missing Exons*.

In the test set there is as much as 37 % of the sequences that result in undefined AC when combining all three programs with the AND method. The number of undefined sequence results is considerably higher for all combinations of programs than shown for the separate programs in Chapter 4.1. The result is understandable since the AND method is much more restrictive and need all the separate programs to predict an exon in a sequence to give the overall prediction of an exon. The number of sequences with undefined AC results has to be taken into account when evaluating the method. The average AC of the sequences for which it was defined is 0.71 for the three combinations of two programs, and 0.71 for the combination of all three programs. The combination of GRAILII and GENSCAN give the highest approximate correlation (0.75/0.73) among the combinations done using the AND method. The results of the best combination can be compared to GENSCAN which gives an AC of 0.77/0.77. There is no combination of the programs using the AND method that improve the approximate correlation.

The GENSCAN and GRAIL combination is also in top when studying the sensitivity among the combinations. When comparing sensitivity with the sensitivity of GENSCAN it is clear that the AND method do not improve this measure either.

Specificity is the proportion of predicted coding nucleotides that are actually coding. The specificity was expected to be improved by the AND method compared to the programs when used separately. The results show that the specificity is improved when studying most of the AND combinations of programs. The combination of all three programs has the highest specificity (0.97) which can be compared to GENSCAN that has a specificity of 0.91 on the test set.

When evaluating sensitivity and specificity together it seems like the GENSCAN and GRAILII combination is on top.

All combinations of programs using the AND method give very high proportion of missed exons which follows from the high portion of predicted non-exonic sequences (i.e. sequences with undefined AC).

The proportion of wrong exons is low for all the combinations, which was expected since the AND method is very restrictive. If specificity and low proportion of wrong exons are important the AND method seems to be appropriate.

4.3 OR

The results of the OR method for combination of gene-finding programs result are shown in Table 4. The results are on both the nucleotide level and the exon level. The first column shows the programs name. The number of sequences with undefined approximate correlation (AC) is shown in the second column (*UNDEF*). These sequences are excluded from the computation of average of the performance measures. *AC* is the average approximate correlation on the nucleotide level. *Sn* is the average sensitivity at the nucleotide level. The average specificity at the nucleotide level is shown under *Sp*. *ME* shows the missing exons, while *WE* shows the wrong exons. The first value in each column is the result using the training set of 146 sequences. The second value in the columns is the result on the test set consisting of 73 sequences.

OR	Undef	AC	Sn	Sp	WE	ME
GR or FE	10/5	0.69/0.66	0.78/0.72	0.77/0.78	0.22/0.22	0.04/0.06
GR or GS	14/12	0.74/0.70	0.83/0.75	0.80/0.81	0.16/0.17	0.04/0.09
GS or FE	7/6	0.71/0.68	0.80/0.74	0.79/0.78	0.19/0.23	0.02/0.08
GR or GS or FE	7/5	0.71/0.68	0.82/0.76	0.76/0.76	0.22/0.24	0.02/0.05

Table 4. The results of the OR combination method. On the nucleotide level: The number of sequences with undefined AC is shown in the second column. The average *AC* (approximate correlation*)*, *Sn* (sensitivity), and *Sp* (specificity) are given for the four possible combinations of programs. Results at the exon level: *WE* is the measure *Wrong Exons*, and *ME* is *Missing Exons*. The combination of GRAILII and GENSCAN is the best when using the OR combination method. The combination has the highest number of undefined ACs, but is on top when looking at the average AC. The combination (GR, GS) has a number of undefined ACs compared to when GENSCAN is used alone, but the AC is worse than the AC that GENSCAN performs alone.

The OR method resulted in lower number of sequences with undefined AC than the AND method, as was expected. 7% of the sequences in the test set resulted in undefined AC for the best OR- combination (i.e. GRAILII or GENSCAN or FEXH) which can be compared to the number of sequences with undefined AC that FEXH gave (11%). The number of sequences without predicted exons is lower for all the combinations of the programs compared to the programs used separately. The combination of GENSCAN and FEXH has almost as low number of sequences with undefined AC as the combination of all three programs. The result shows that even FEXH contributes to an improved result.

The average AC is 0.68 for the three possible combinations of two programs, and the same (0.68) for the combination of all three programs. The best combination of the programs when using the OR method is the combination of GENSCAN and GRAILII when considering AC. The AC of the combination of GS and GR is 0.70 for the test set which can be compared to the AC of GENSCAN when used alone (0.77). None of the combinations of programs using the OR method improves the approximate correlation compared to GENSCAN when used alone.

The combination of GRAILII and GENSCAN scores the best sensitivity and specificity among the OR-combinations. When looking at only the test set results the combination of GRAILII and GENSCAN has the highest specificity, while the combination of all three programs has the highest sensitivity. All the combinations using the OR method improve the sensitivity measure compared to when the programs are used separately and the AND method. It was expected that the sensitivity would be improved by the OR method and this was also shown by Murakami and Takagi (1998).

The proportion of wrong exons is somewhat higher for the OR method compared to when the programs are used separately and considerably higher than for the AND method. It was expected that the OR method would have more wrong exons than the AND method because of the prediction generosity.

The portion of missing exons is lower than for the AND method and for the programs when used separately. This is also a result of the prediction generosity of the OR method which was also seen in the nucleotide level results.

4.4 MAJORITY

The results of the MAJORITY method for combination of gene-finding programs results are shown in Table 5. The results are on both the nucleotide level and the exon level. The first column shows the program name. The number of sequences with undefined approximate correlation (AC) is shown in the second column (*UNDEF*.). These sequences are excluded from the computation of average of the other measures. *AC* is the average approximate correlation on the nucleotide level. *Sn* is the average sensitivity at the nucleotide level. The average specificity at the nucleotide level is shown under *Sp. ME* shows the missing exons, while *WE* shows the wrong exons.

The first value in each column is the result using the training set of 146 sequences. The second value in the columns is the result on the test set consisting of 73 sequences. The results have been divided in two (training and test) to show differences in the training and test sets and also to enable the comparison to the ANN results. The ANN results are only shown for the test set.

MAJ	Undef	AC	Sn	Sp	WE	ME
GR,GS,FE	22/16	0.77/0.73	0.77/0.71	0.90/0.91	0.08/0.09	0.05/0.10

Table 5. The results of the MAJORITY combination method used with the three gene-finding programs. On the nucleotide level: The number of sequences with undefined AC is shown in the second column. The average *AC* (approximate correlation), *Sn* (sensitivity), and *Sp* (specificity) are given for the four possible combinations of programs. Results at the exon level: *WE* is the measure *Wrong Exons*, and *ME* is *Missing Exons*.

11 % of the sequences in the test set resulted in undefined AC. The average AC of the remaining sequences in the test set is 0.73. The MAJ method result in a much lower number of undefined ACs than the AND method, but somewhat higher than the corresponding number for the OR method.

The resulting AC of the MAJ method is higher or comparable than all possible combinations using either the AND and OR method, which was expected. There are two combinations of two programs using the OR method (i.e. GRAIL or GENSCAN, GRAIL or FEXH) that resulted in the same AC as the MAJ method. The MAJ method has somewhat lower AC on the test set compared to GENSCAN, while on the training set it is comparable to GENSCAN.

The sensitivity of the MAJ method is higher than the sensitivity for the AND method and lower than the sensitivity for the OR method.

The MAJ method gives lower specificity than the AND method and higher specificity than the OR method. In fact the specificity of the MAJ method is higher than the average between the AND and OR methods when combining all three programs.

It was expected that the MAJ method would give sensitivity and specificity between the AND and OR methods.

When looking at the wrong exons the MAJ method has a WE comparable to some of the AND combinations (e.g. GR and FE) and thereby lower that the programs used separately.

The Missing exon measure is improved by using the MAJ method compared to the when the programs are used separately. The ME is comparable to the combinations using the OR method.

4.5 ANN type 1

The results of the ANN type 1 are shown in Table 6. The results shown for the combinations of two programs are after 9000 epochs of training. Three input nodes and more hidden nodes result in more weights to train for the network. Intuitively the combination of all three programs has more information about the sequence and it needs more training to approximate a suitable function. The result of the combination of the three programs shown in Table 6 is the result after 18000 epochs of training. It is the result of test 5 that is shown here. Because of time limits in this project and limits of computer capacity the ANN combinations of two programs could not be trained for more epochs. The mean square error is still improving after 9000 epochs and the tests of the

ANNs show improving results. This suggests that more training would give even better results. The results shown in Table 6 are the results of using the trained ANN on the test set. The results are on both the nucleotide level and the exon level. The first column shows the program name. The number of sequences with undefined approximate correlation (AC) is shown in the second column (*UNDEF*.). These sequences are excluded from the computation of average of the other measures. *AC* is the average approximate correlation on the nucleotide level. *Sn* is the average sensitivity at the nucleotide level. The average specificity at the nucleotide level is shown under *Sp*. *ME* shows the missing exons, while *WE* shows the wrong exons.

The first value in each column is the result using the training set of 146 sequences. The second value in the columns is the result on the test set consisting of 73 sequences. The results have been divided in two (training and test) to show differences in the training and test sets and also to enable the comparison to the ANN results. The ANN results are only shown for the test set.

The results shown here indicate the possibilities of using even the simplest ANN for this problem.

ANN	Undef	AC	Sn	Sp	WE	ME
GR, GF (test 1)	17	0.68	0.64	0.88		
GR, GS (test 2)	19	0.77	0.75	0.92		
GS, FE (test 3)	19	0.77	0.87	0.74		
GR, GS, FE (test 5)	22	0.85	0.99	0.80	0.01	0.11

Table 6. The results of the ANN type 1 combination method. On the nucleotide level: The number of sequences with undefined AC is shown in the second column. The average *AC* (approximate correlation*)*, *Sn* (sensitivity), and *Sp* (specificity) are given for the four possible combinations of programs. Results at the exon level: *WE* is the measure *Wrong Exons,* and *ME* is *Missing Exons*.

The ANN type 1 combination results in 23-30% sequences with undefined AC out of the total number of sequences in the test set. This is higher than when the programs are used separately (11-25%), but not as high as the AND method (30-37%).

The highest average AC is reached by the combination of all of the three programs. The results indicate that this method possibly can improve the AC, as more programs are included in the combination. The average AC of the ANN combination of all of the three programs (0.85) can be compared to the average AC of the best performing program (i.e. GENSCAN) when used alone (0,77). The ANN method is the only method included and evaluated in this project that clearly improves the approximate correlation when combining gene-finding programs.

The sensitivity for the combination of all three programs is 0.99 which can be compared to the sensitivity of the OR combination of all three programs (0.76) and GENSCAN (0.77). The ANN shows an considerable improvement in sensitivity.

The highest specificity is scored by the combination of GRAILII and GENSCAN (0.92). The average of specificity for the ANN combinations of two programs is 0.85, while the combination of all three programs is 0.80. The resulting specificity can be explained by the higher number of sequences with undefined AC resulting from the combination of all three programs.

The proportion of wrong exons is 0.01 for the combination of all three programs. This is the lowest WE result compared to all other combinations and the programs used separately. The proportion of missing exons is lower than shown for the separate programs, but not as low as for the OR method.

When evaluating these results it has to be kept in mind that the ANN can be trained more and it can improve the prediction accuracy even more. The results for the AND, OR, and MAJ methods are final.

4.6 ANN type 2

Experiments have been done using the ANN type 2, but because of limits in this project these results cannot be shown here. More work will be done in the next couple of months using the ANN type 2 for combining gene-finding program results.

4.7 Chapter summary

The results of the experiments done within this project are presented in this chapter. When the programs are used separately GENSCAN is the best performing program when measured by all the measures used in this project. On the test set GENSCAN results in an approximate correlation of 0.77. The sensitivity is 0.77 and the specificity is 0.91. On the exon level the missing exon is 0.09 and the wrong exon is 0.09. FEXH is the worst performing program used in this project with an approximate correlation of 0.62, a sensitivity of 0.63, and a specificity of 0.80. On the exon level FEXH gives the highest rate of wrong exons, 0.22, but GRAILII is the worst program when considering the rate of missing exons (0.18).

The combination of GRAILII and GENSCAN and the combination of GENSCAN and FEXH are the two combinations of programs that perform best when using the AND method. The two combinations (GR and GS, GS and FE) have the approximate correlation 0.73. When comparing the rate of wrong exons and the specificity the combination of all three programs is the best performing combination when using the AND method. The rate of missing exons on the other hand is the highest for this combination (GR and GS and FE) among all the combinations using the AND method. As expected and understandable the AND method results in few predictions, while the probability of a prediction being actually true is very high.

When using the OR method the combination of GRAILII and GENSCAN performs the best if considering the approximate correlation (0.70), the specificity (0.81), and the rate of wrong exons (0.17). When adding FEXH to the combination the approximate correlation, the specificity, and the rate of wrong exons decreases, while the sensitivity and the rate of missing exons are improved. The results show that FEXH is able to find some exons that the others do not and that FEXH predicts wrong exons that the others do not.

The combination method MAJ shows results that are somewhere in between the results of the AND and OR methods. The approximate correlation of the MAJ method is 0.73 on the test set, which is somewhat worse that the approximate correlation of the best performing separate program (GENSCAN). When looking at the approximate correlation for the training set the MAJ method performs as well as GENSCAN.

Results of the simplest ANN show improved results compared to the logical methods as well as compared to all the participating programs when used separately and when considering approximate correlation, sensitivity, wrong exons and missing exons. It has to be kept in mind that the ANN can be trained even more to perform even better. Because of limits to this project the ANN type 1 combining GRAILII, GENSCAN and FEXH predictions could not be trained for more than 18000 epochs.

The hypothesis stated in this project was that an ANN would give better results than the logical methods (i.e. AND, OR, MAJ) when used to combine gene-predictions done by different programs. The results shown in this chapter fulfill the hypothesis.

5.0 Analysis

The results presented in Chapter 4 are analysed in this chapter on both the nucleotide level and exon level.

The aim of this project was to evaluate methods for combination of results from gene-finding programs. It is interesting to study the differences between the methods as well as how the methods are affected by an increasing number of programs combined. The performance measures approximate correlation, sensitivity, specificity, missing exons, and wrong exons are used for the evaluation of the methods. For comparison the best performing program is included in the analysis using the different measures. Graphs in this chapter show how the number of programs combined with a method affect the performance measures and thereby the prediction accuracy. In the following sections the values for one and two programs are the average of values for all three programs when used separately and the average of all three possible combinations of two programs. The ANN method that is presented in the graphs is the ANN type 1.

The average approximate correlation for the methods is shown in Figure 18. The graphs in Figure 18 illustrate the difference in AC between the methods and how the average AC is affected by the number of gene-finding programs included. The graph shows clearly that the ANN method is the most promising one in the set of methods evaluated in this project. The AND, OR, and MAJ methods have lower approximate correlation than GENSCAN which is the best performing program when used separately. The ANN reaches an AC of 0.85, which can be compared to the AC of GENSCAN shown in here (0.77).

The AC is improved even after a third program is included in the ANN combination. This shows that even the worst performing program in this project has some gene-finding features that the others do not have and that these features are taken advantage of in the overall prediction. Murakami and Takagi (1998) expressed the idea that a combination of programs would improve the prediction accuracy. They also discussed the possibilities of capturing the advantages of a program that finds some gene-features that others do not have, but perhaps performs worse in the general case. The results in this project show that the ANN method somewhat capture the best of all programs and improved the prediction accuracy markedly.

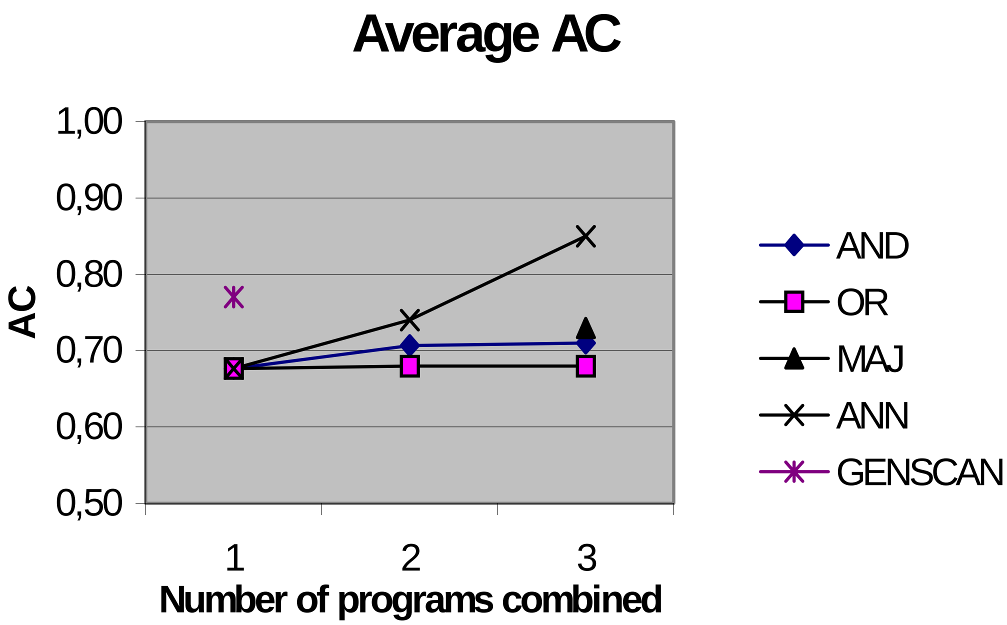


Figure 18. The average AC of the different methods affected by the number of programs combined. The AND and OR methods do not improve the AC markedly when combining two programs compared to when the programs are used separately. As the number of programs combined increases (i.e. from two to three) the AC of the AND and OR method stop improving. The MAJ method improves the AC somewhat compared to the average of the separate programs. The AND, OR and MAJ methods perform worse than GENSCAN which is the best performing program. The ANN type 1 continues to improve average AC as the number of programs combined increases. The best performing (i.e. highest average AC) program is also included in the figure to illustrate which methods improve the average AC. The results show that it is only the ANN that clearly improves the average AC (when combining all three programs) when comparing the combinations to GENSCAN.

Figure 19 shows the resulting sensitivity for the different methods and how it is affected by an increasing number of combined programs using the different methods. The results show that the ANN method improves the sensitivity as the number of combined programs increase. As expected the sensitivity of the AND method decrease when combining two or three methods, while the same measure is improved by the OR method. The MAJ method has a sensitivity value in between the AND and OR methods, which was also expected. It is only the ANN combination of all three programs that has considerably higher sensitivity than GENSCAN even though the ANN combination of two programs is on average higher than the average sensitivity of the separate programs.

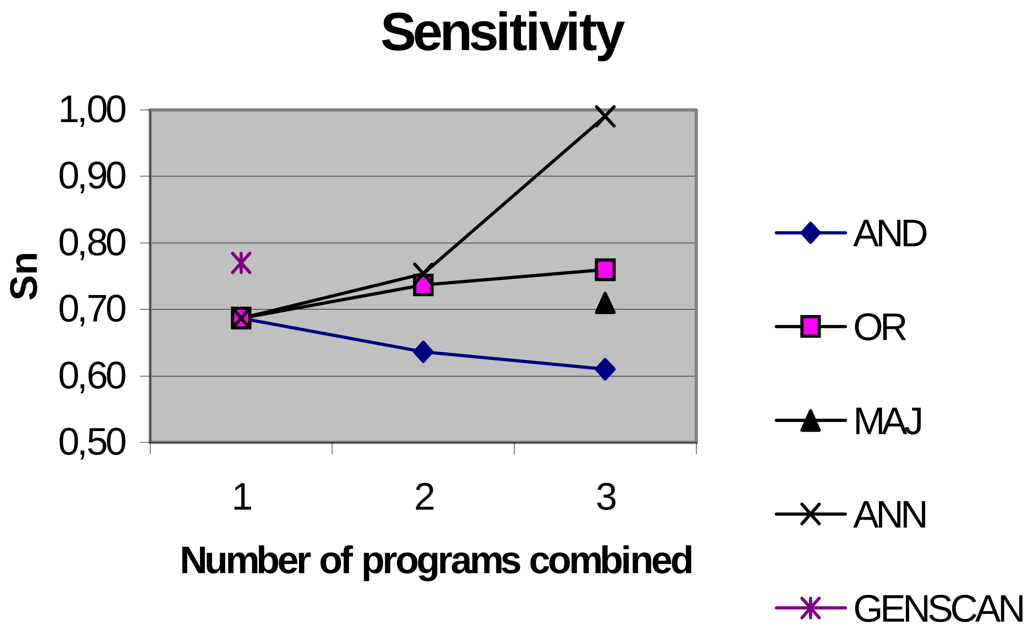


Figure 19. Nucleotide level results when measuring sensitivity. Sensitivity is the portion of actual coding nucleotides that are correctly predicted as coding. The sensitivity gets worse when combining programs using the AND methods. The OR and the MAJ methods improve the sensitivity somewhat when combining two and three programs compared to the average sensitivity of the separate programs. Compared to GENSCAN the MAJ and OR methods do not improve the sensitivity at all. It is only the combinations with the ANNs that clearly improve the sensitivity when combining two and three programs. The results demonstrate that the sensitivity is improved when increasing the number of programs combined with an ANN. All methods have been compared to the best performing programs when used separately (i.e. GENSCAN).

In Figure 20 the specificity of the different methods are shown and the affect of combining two or three programs using the different methods. As expected the AND method improves the specificity as the number of combined programs increases. The combinations of two programs using the AND methods improve the specificity compared to GENSCAN. It was also expected that the OR method would decrease the specificity as the number of programs increased. The MAJ method has a specificity value between the AND and OR methods as expected. The specificity of the MAJ method is comparable to the specificity of GENSCAN. The ANN improves the specificity, as two and three programs are included in the combination compared to the average specificity of the separate programs. The result shows a somewhat comparable specificity of the ANN using three programs and GENSCAN. It has to be kept in mind that the ANN can be trained to even better accuracy in terms of output error and perhaps improve the specificity even more.

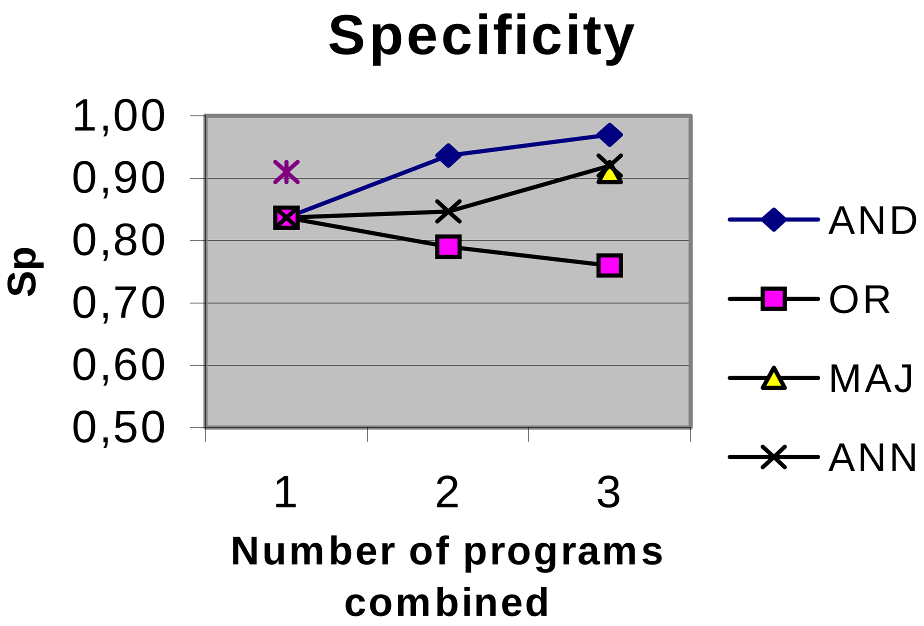


Figure 20. The specificity of the different methods and the affect of the increasing number of programs combined. The AND method shows the highest specificity and the OR method shows the lowest. The MAJ method performs better than the average between AND and OR. The ANN does not have as high specificity as the AND method, but it has to be kept in mind that the ANN can be trained more. The AND, MAJ, and ANN methods all improve specificity as the number of programs combined increase and they perform better than the best individual program (i.e. GENSCAN).

In Figure 21 the number of sequences resulting in undefined AC for the different method are shown and the effect of the increasing number of programs combined using the methods. A sequence that results in an undefined approximate correlation is a sequence that has no true positives and no false negatives, or no false positives and no true negatives, or no true positives and no false positives, or no false negatives and no true negatives. Since it is common that a prediction result in no positives at all the number of sequences with undefined approximate correlation can be a problem that has to be considered in the evaluation of the program or the combination method.

Because of the nature of the AND method the number of sequences with no positive prediction increase as the number of programs combined increase. The OR method naturally result in a lower number of sequences with undefined AC as the programs combined increase. Compared to FEXH it is only the OR combination of all three programs that has a lower number of undefined ACs. The MAJ method has results that are between the results of the NAD and OR method. The ANN method has a high number of sequences with undefined AC. Especially the ANN combination of three programs has a high number of undefined ACs. It has to be kept in mind that the ANN can be trained more and possibly increase the number of sequences that include positive predictions. It is a question of how strong the predictions are in the sequences.

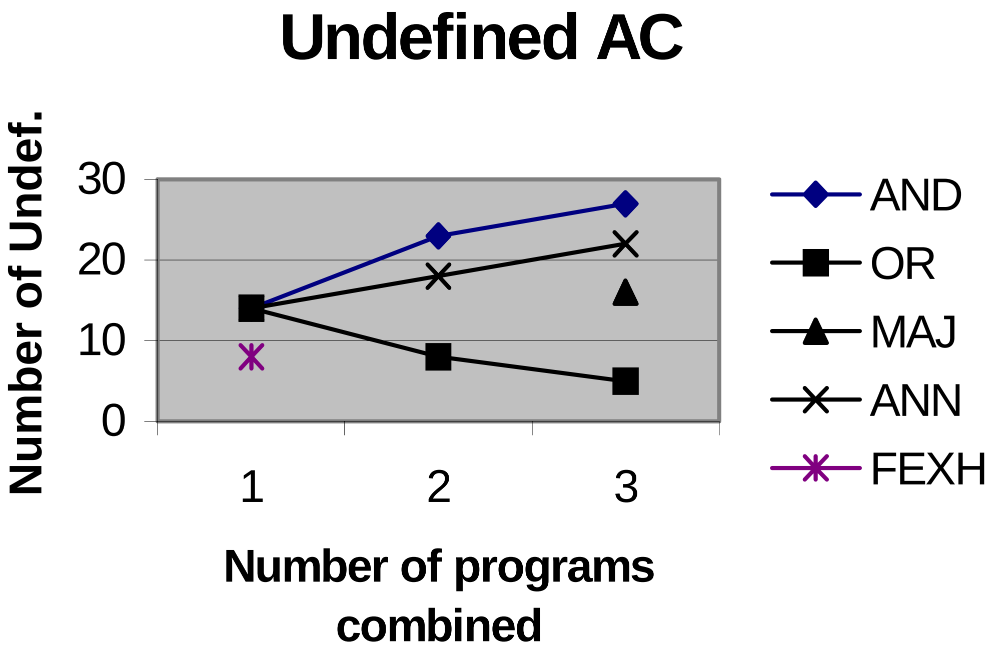


Figure 21. The average number of undefined ACs at the nucleotide level. These are sequences which are predicted to contain only coding nucleotides or only non-coding nucleotides, or which actually contain only coding nucleotides or only non-coding nucleotides. The AND method increases the number of sequences with undefined ACs as the number of combined programs increases. The OR method has the lowest number of sequences with undefined AC. The MAJ method give a number of sequences with undefined AC that are comparable lower than that of the average of the AND and OR methods. Both the ANN method and the OR method result in less undefined AC than the best performing program FEXH.

The results measured by the missing exons are shown in Figure 22 for the defined sequences. The results of the different methods are shown as well as the result of GENSCAN wich is the best performing single programs when considering the missing exons. The missing exon measure shows the sensitivity at the exon level.

The ANN increase the ME value as the number of combined programs increase, which was expected because of the restrictive nature of the method. The OR method improves the ME value as soon as only two programs are combined and the method improves the ME as a third program is included. The low rate of missing exons performed by the OR method is understandable since it includes all predictions done by all programs in the overall prediction. The AND method has a rate of missing exons that is comparable to that of GENSCAN alone. As mentioned earlier it has to be kept in mind that the ANN can be trained more and possibly improve the rate of missing exons.

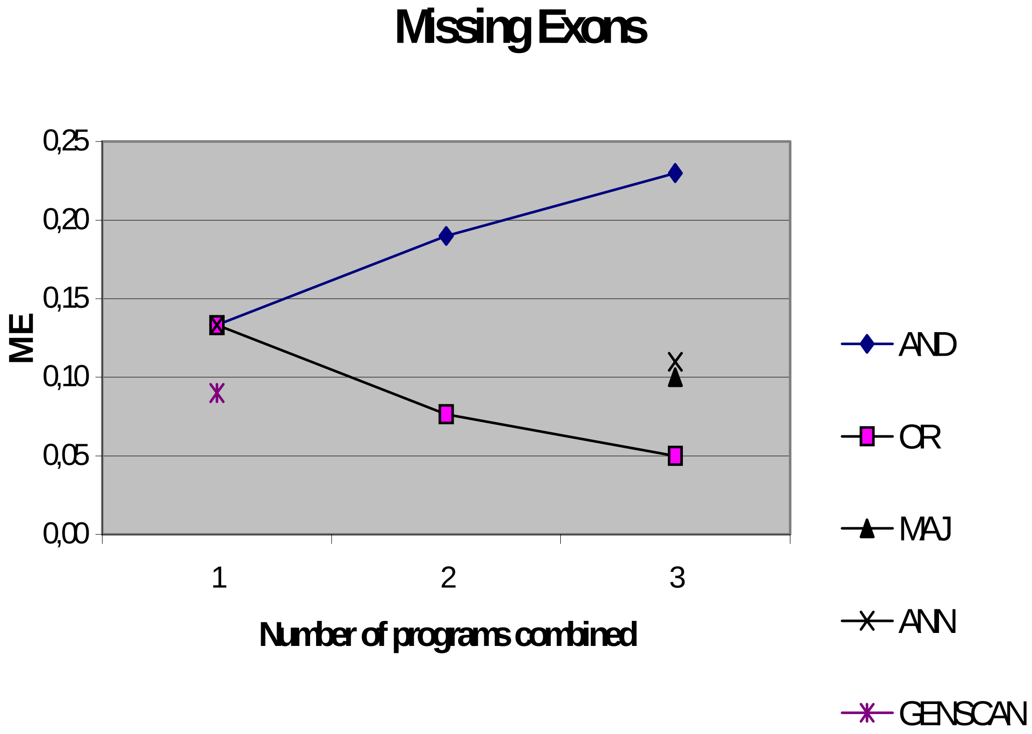


Figure 22 The portion of missing exons. The results of the best performing program (i.e. GENSCAN) is shown for comparison. The graphs illustrate the effect on the combination methods of the increasing number of programs combined. The OR method has the lowest rate of missing exons, which was expected. The AND method results in higher ME value as the number of programs combined increase. The MAJ method result in a value between the AND and OR methods. The result of the ANN is comparable to that of GENSCAN, but not as good as that of the OR method.

In Figure 23 the rate of wrong exons is shown for all combination methods and for the best performing single program. The result show that the rate of wrong exons predicted is lowest when combining all three programs with the ANN. It is a considerable improvement when combining two and three programs using the AND method compared to GENSCAN and the average of the single program. The result of the OR method was expected. The OR method includes more wrong exons in the overall prediction as the number of programs combined increase. The MAJ method have a rate of wrong exons between the AND and OR methods and it is comparable to that of GENSCAN alone.

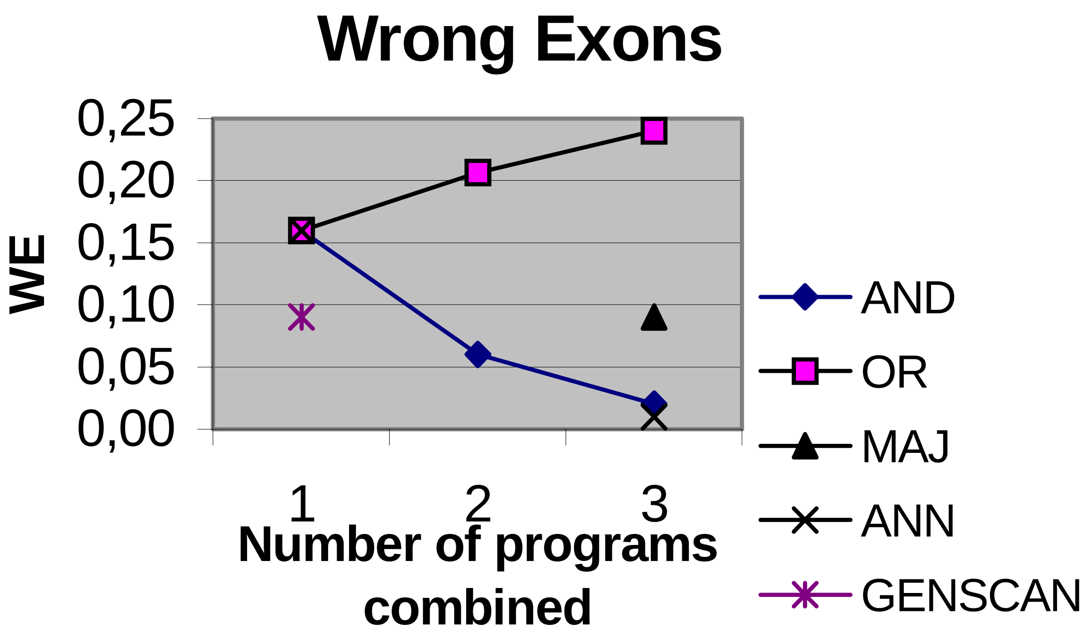


Figure 23. The rate of wrong exons. The result of GENSCAN is shown for comparison. The OR method shows the highest rate of wrong exons, which follows from the generous nature of the method. The AND method shows the lowest rate of wrong exons among the logical methods, which was expected. The MAJ method has a rate that is comparable to the average of the AND and OR method. The result of the ANN is somewhat better than the AND method when combining the three programs.

5.1 Chapter summary

There is a clear indication that simple logic-based combinations are not entirely suitable for combinations of predictions. A prediction is not always discretely coding or non-coding. The prediction weight, score, or probability reflects the prediction certainty.

The idea of combining different gene-finding programs was that different programs might be better at finding different types of exons or better at finding exons in different sequence structures. If the aim is to capture these valuable differences between the programs it is obvious that combinations of the programs using the AND method would fail. With the AND method only the predictions made similar by all programs will be considered and which is shown in this project, this results in a large number of missing exons in the overall prediction. The OR method is not sufficient for the prediction combination problem as long as the programs predict wrong exons. With the OR method any program can predict an exon anywhere in the sequence for the overall prediction to include that exon. This results in very few missing exons, but a large number of incorrectly predicted ones are included.

As shown by the results in this project there are functions more appropriate than the logical function (i.e. AND, OR, and MAJ) for the problem of combining predictions. The simple ANNs used in this project have approximated functions for this problem that give a more correct overall prediction than AND, OR, and MAJ. The AC of the best performing program (i.e. GENSCAN) when used separately was 0.77 and the number of sequences with undefined AC was 18 for the test set. The GENSCAN results can be compared to the ANNs AC when combining all three programs. The AC of the ANN after 18000 epochs of training when combining three programs was as high as 0.85 and the number of undefined sequence results was 22.

9 Discussion

This chapter aims at discussing details in the project that has affected the results.

The prediction outputs from the separate programs are crucial to the whole project. If all three of the programs fail to predict exons in a sequence the sequence cannot get an improved overall resulting prediction after using any of the combination methods. It is interesting to look into if there are sequences that have no predicted exons in the data set. The particularities of these sequences should be studied and perhaps a fourth program can be found that finds these exons. With the ANN method it is very easy to include more programs in the combinations.

The program predictions were normalized and the sequence encoded using 0.2 for introns and the normalized scores of the predicted exons were used to encode the coding regions. Because of the wide range of weights that could be assigned to prediction by FEXH, the average deviation was used to find values that corresponded to 0.2 and 0.8. In the process of normalizing FEXH weights there were a few weakly predicted exons that were to weak to be considered. It is possible that the weakest predicted exons actually overlap with actual exons. There might be a more appropriate way of normalizing the FEXH weights. On the other hand FEXH showed the highest rate of incorrectly predicted exons and the lowest approximate correlation. If considering even the weakest predictions it is most likely that the rate of wrong exons will rise even more.

The logical methods AND and OR used in this project are inspired by the AND and OR methods used by Murakami and Takagi (1998). In this project all prediction done by all programs after normalization are considered when applying the methods. In the experiments of Murakami and Takagi (1998) the weights and scores assigned to prediction by the programs were transformed to a probability using the training set of sequences. Murakami and Takagi (1998) used the training set of sequences to find a threshold for which the overall prediction probabilities were measured after using the combination methods. A threshold would cut the weakest predictions that are least likely to overlap with actual exons. In this project a threshold is not used and the weights and scores assigned to predicted exons by the programs are not transformed to probabilities. It is questionable if it is justifiable to transform the scores and weight into probabilities. Since Murakami and Takagi (1998) used a threshold for the overall predictions, their results do not agree exactly with the results in this project. The overall behavior of the methods shown in this project can be compared with the results shown by Murakami and Takagi (1998) (e.g. the AND and OR are the two methods that are shown to be the two extremes. The AND method show high specificity and low sensitivity, while the OR method shows lower specificity and high sensitivity).

As discussed above the prediction quality of the separate programs is crucial. If programs do not predict exons in a sequence, even though there are actual exons, the approximate correlation is not defined. A resulting prediction that fail to predict some exons is a far from sufficient. It is common that one or two programs miss exons and the approximate correlation is not defined. Sequences for which the approximate correlation is not defined are excluded from the calculation of the performance measures. It is a problem that sequences with undefined approximate correlation is excluded from the performance evaluation. The results from an evaluation that do not consider the number of sequences with undefined approximate correlation are misleadingly good.

The length of the sequences in the data set could possibly have an affect on the result. Different gene-finding programs might not work similarly well on very short or very long sequences. If none of the programs in a data set can recognize exons in very short sequences, for instance, this will affect the result even after the combination of programs. As said before the performance of the separate programs is a very important issue even when good combination methods are used.

The ANN methods are time consuming to train. Because of limits to this project and limits in the computer capacity the ANN type 1 combination of two programs result after 9000 epochs and the ANN type 1 combination of three programs are shown in this dissertation. Further work with the ANN type 2 will be done after completion of this dissertation.

The threshold used for separating predicted coding and predicted non-coding nucleotides by the ANN was initially set to 0.5. It is possible that a more appropriate threshold can be found that will result in a even more improved accuracy.

Hypothetically, the function approximated by the ANN could be extracted from the network using methods for rule extraction proposed by e.g. Andrews et. al. (1995) and Tickle et. al. (1997). Rule extraction is not a part of this project.

The exons documented for the sequences in the GenBank files are used as the actual exons in this project. If there are incorrect exons documented in the GenBank files the problem follows through the entire project.

7 Conclusions

The conclusions drawn from the results of the experiments in this project are presented in this chapter.

The aim of the project was to evaluate five methods for combining the results of gene-finding programs. The hypothesis of the project was that a machine learning method such as an ANN would improve the average AC more than the logical methods AND, OR, and MAJ. The results from the experiments presented in Chapter 4 fulfill the hypothesis.

The results shown in this project demonstrate that the prediction performance can be improved when combining a number of different gene-predicting programs compared to when the programs are used separately. It is the method for combining the programs that need consideration. These conclusions agree with the conclusions drawn by Murakami and Takagi (1998).

The simplest forms of logical functions (e.g. AND, OR, MAJ) for combining prediction results do generally improve the prediction performance some. This conclusion agrees with conclusions drawn by Murakami and Takagi (1998).

The results of the ANNs used in this project improve the prediction performance considerably compared to when the programs are used separately. There is also a considerable higher performance compared to the logical functions AND, OR, and MAJ. The conclusion drawn from this is that there are functions more appropriate for the combination of the predictions than AND, OR, and MAJ.

The ANN method is the only evaluated in this project that clearly improves the prediction accuracy as the number of programs combined increase. The ANN method is also the method with the highest AC when three programs are combined.

Although the ANN method proved to be the best performing combination method (when considering AC) used in this project, there still remains work to be done with respect to finding the optimal artificial neural network for this task. The optimization of ANNs for combination of gene-finding programs is left as an open issue for researchers to investigate.

10 Future work

This chapter aims at proposing some interesting follow-up projects related to this project.

The results in this project indicate that the logical methods for combining results from prediction-programs like gene-finding programs are not the most promising ones. The results showed by the simple ANNs used in this project improve the prediction accuracy from 0.77 to 0.85 compared to when the programs are used separately. Since there are many possible improvements to these ANNs, it is very likely that the performance of the ANNs for this problem can be improved much more.

The problem domain focused on in this project was gene-finding. Prediction-programs like the gene-finding programs used here are found in other areas of bioinformatics and other sciences as well. It would be interesting to apply the combination methods used here to another domain (e.g. protein folding prediction). This would also show the differences between the methods when using them to combine predictions. The results would probably agree with the results shown in this project.

Acknowledgements

I would like to thank my supervisors Dan Lundh and Björn Olsson for their constructive and patient help during the six months of this project. Your careful and detailed criticism has improved this dissertation allot.

Bibliography

Andrews, R., Diedrich, J., Tickle, A.B. 1995. Survey and critique of techniques for extracting rules from trained artificial neural networks. *Knowledge Based Systems* 8: 373-389.

Burge, C., Karlin, S. 1997. Prediction of Complete Gene Structures in Human Genomic DNA. *Journal of Molecular Biology* 268: 78-94.

Burset, M., Guigo, R. 1996. Evaluation of Gene Structure Prediction Programs. *Genomics* 34: 353-367.

Craven, M.W., Shavlik, J.W.1994. Machine learning Approaches to gene recognition. *IEEE Expert.*

Fickett, J.W. 1996. Finding Genes by Computer: the state of the art. Trends in Genetics 8: 316-320.

Fields, J.W., Soderland, C.A. 1990. gm: a practical tool for automating DNA sequence analysis. *CABIOS*, 6(3):263-270.

Gelfand, M.S., Mironov, A.A., Pevzner, P.A. 1996. Gene recognition via spliced sequence alignment. *Proceedings of the National Academy of Sciences of the USA* l93:9061-9066.

Giegerich, R., Meyer F., Schleiermacher C. 1996. Genefisher-software support for the Detection of postulated genes. *Proceedings of the Fourth International Conference on Intelligent Systems for Molecular Biology*. 68-77. AAAI Press.

Griffiths, A.J.F., Miller, J.H., Suzuki, D.T., Lewontin, R.C., Gelbart, W.M. 1996. *An introduction to Genetic Analysis.* 6 ed. W.H. Freeman Company, New York.

Harris, N.L. 1997. Genotator: A workbench for sequence annotation. *Genome research* 7: 754-762.

Kleinsmith, L.J., Kish, V.M. 1995. *Principles of Cell and Molecular Biology*. 2 ed. HarerCollins College Publishers, New York.

Krogh, A. 1997. Two methods for improving performance of an HMM and their application for gene finding. Gaasterland, T. Et. al. Eds. *Proceedings of Fifth International Conference on Intelligent Systems for Molecular Biology*. 179-186. AAAI Press.

Kulp, D., Haussler, D. 1997. Integrating database homology in a probabilistic gene structure model. *Pacific Symposium on Biocomputing 97*. 232-244.

Kulp, D., Haussler, D., Reese, M.G., Eeckman, F.H. 1996. A Generalized Hidden Markov Model for the Recognition of Human Genes in DNA. *Proceedings of the Fourth International Conference on Intelligent Systems for Molecular Biology*, 134-142. AAAI Press.

Lopez, R., Larsen, F., Prydz, H. 1994. Evaluation of the exon prediction of the GRAIL software. *Genomics* 24:133-136.

Murakami K., Takagi T., 1998. Gene recognition by combination of several gene-finding programs. *Bioinformatics* 8:665-675.

Mural, R.J., Einstein, J.r., Guan, X., Mann, R.C., Uberbacher, E.C. 1992. An artificial Intelligence Approach to DNA Sequence Feature recognition. *Trens in Biotechnology*, 10:66-69.

Rogozin, I.B., D'Angelo, D., Milanesi, L. 1999. Protein-coding regions prediction combining similarity search and conservative evolutionary properties of protein-coding sequences. Gene 226:129-137.

Russell, S., Norvig, P. 1995. *Artificial intelligence- A modern apporach*. Prentice Hall Intern Ed. New Jersey.

Salzberg, S.L. 1997. A method for identifying splice sites and translational start sites in eucaryotic mRNA. *Cabios*, 4:365-376.

Salzberg, S.L., Delcher, A.L., Kasif, S., White, O. 1998. Microbial gene identification using interpolated Markov models. *Nucleic Acids Research*. 2:544-548.

Salzberg, S., Chen, X., Henderson, J., Fasman, K. 1996. Finding genes in DNA using decision trees and dynamic programming. *Proceedings of the Fourth International Conference on Intelligent Systems for Molecular Biology* 201-210.AAAI Press.

Singer, M., Berg P. 1991. *Genes & Genomes*, University Science Books, Mill Valley California.

Smith, R.F., Wiese, B.A., Wojzynski, M.K., Davison, D.B., Worley, K.C. 1996. BCM search launcher- An integrated interface to molecular biology data base search and analysis services available on the worl wide web. *Genome research* 6:454-462.

Snyder, E.E., Stormo, G.D. 1995. Identification of protein coding regions in genomic DNA. *Journal of Molecular Biology*, 248:1-18.

Snyder, E.E., Stormo, G.D. 1993. Identification of coding regions in genomic DNA sequences: ans application of dynamic programming and neural networks. *Nucleic Acids Research*, 3:607-613.

Snustad, D., Simmons, J,. Jenkins, J.B. 1997. *Principles of Genetics*, John Wiley & Sons, Inc.USA.

Solovyev, V.V., Salamov, A. The Gene- Finder computer tools for analysis of human and model organisms genome sequences. *Proceedings of the Fifth International Conference on Intelligent Systems for Molecular Biology*. 294-302. AAAI Press.

Solovyev, V.V., Salamov, A.A., Lawrence, C.B. 1994. The prediction of human exons by oligonucleotide composition and discriminant analysis of spliceable open reading frames. Altman R., Brutlag D., Karp P., Lathrop R., Searls D. eds. *Proceedings of the Second International Conference on Intelligent Systems for Molecular Biology*. 354-362.

Sternberg, M.J.E. 1996. *Protein structure prediction- A practial approach*. Oxford University Press. Oxford.

Strachan T., Read A. P, *Human Molecular Genetics*, BIOS Scientific Publishers Limited, Oxford UK, 1997.

Tickle, A.B., Andrews, R., Golea, M., Diedrich, J. 1997.*Rule extraction from trained artificial neural networks*. TOP Publishing.

Uberbacher, E.C., Einstein, J.R., Guan, X., Mural, R.J. 1994. Gene recognition and assembly in the GRAIL system: progress and challenges. Lim, H.A., Fickett, J.W., Cantor, C.R., Robbins, R.J. eds. *Proceedings of the Second International Conference on Bioinformatics, Supercomputing and Complex Genome Analysis*. 465-467. World Scientific.

Uberbacher, E.C., Mural, R.J. 1991. Locating protein-coding regions in human DNA sequences by multiple sensor-neural network approach. *Proceedings of the National Academy of Sciences of the USA*. 88:11261-11265.

Xu, Y., Einstein, J.R., Mural, R. J. 1994. An improved system for exon recognition and gene modeling in human DNA sequences. *Proceedings Second International Conference on Intelligent Systems for Molecular Biology*. 376-384. World Scientific.

Xu, Y., Uberbacher, E.C. 1996. Gene prediction by pattern recognition and homology search. *Proceedings of the Fourth International Conference on Intelligent Systems for Molecular Biology*. 241-251. AAAI Press.

Xu, Y., Einstein, J.R., Mural, R.J., Shah, M., Uberbacher, E.C. 1994. An improved system for exon recognition and gene modeling in human DNA sequences. *Proceedings Second international conference on intelligent systems for molecular biology*. 376-384. World Scientific.

Xu, Y., Uberbacher, E.C. 1997. Automated gene identification in large-scale genomic sequences. *Journal of Computational Biology*, 3:325-338.

Zhang, M.Q. 1997. Identification of protein coding regions in the human genome by quadratic discriminant analysis. *Proceedings of the National Academy of Sciences of the USA.* 94:565-568. Genetics. USA.

Appendix A

Loci	Lenght	Loci	Lenght	Loci	Lenght
D49493	17286	HSMHC3W36A	100267	HSU58975	1161
D50001S17	282	HSMOGG	17538	HSU59227	7451
D63789	5669	HSMRCOXA	210	HSU60232	1210
D63790	5660	HSNADGLCT	6743	HSU60289	4742
D82881	339	HSNF1GEN01	244	HSU60871	150
D86566	19654	HSNF1GEN12	419	HSU61148	1572
D89501	7201	HSNF1GEN19	805	HSU62025	3541
HS179D3B	31330	HSNPYY2S2	3444	HSU62556	2608
HS326L13	127247	HSP137GPI	244	HSU63329	1869
HSADH41	515	HSP65D	1473	HSU63842	1268
HSARS81S	2063	HSPEX1	960	HSU66083	73360
HSATIH101	1392	HSPLK3	5389	HSU66109	185
HSBDKRBI2	3332	HSPPAE1	813	HSU66711	5543
HSBKLA	838	HSPRKAR2A	1956	HSU66840	2166
HSBRCA1	1482	HSPRMTNP2	16851	HSU70137	921
HSC2REG	1557	HSPRPEX18	636	HSU71086	4030
HSCAM3X1	1863	HSPTHR05	1980	HSU74651	8699
HSCATG1	632	HSSAGART15	215	HSU76619	5331
HSCCNC12	137	HSSP100GN	1260	HSU77737	2974
HSCD89EX1	1154	HSTHYR1	350	HSU77827	1648
HSCGT05	794	HSTHYR13	407	HSU79549	151193
HSCKRL1	1158	HSTPRCGEN	2053	HSU80055	1709
HSCLN3	15997	HSU20325	2483	HSU80982	2536
HSCOX6BL	1458	HSU24186	1565	HSU82609	2878
HSECE11X2	790	HSU26593	592	HSU83908	1740
HSECMP6	300	HSU31767	3973	HSU85035	335
HSENOYL1	1942	HSU31929	8851	HSU91934	2267
HSEPC1EX7	552	HSU32672	1535	HSUSF2	14440
HSF62D4B	22007	HSU32674	1293	HSVASPEX1	2494
HSFGF8S4	402	HSU34804	8397	HSXBXVIII	12700
HSG6PDGEN	52173	HSU34806	1232	HUM17BHSDI	4194
HSGCAP2	3600	HSU37106	3186	HUM2G3A	110858
HSGLBN	11376	HSU40391	3453	HUMALAD13	4704
HSGLUD113	1587	HSU41290	1308	HUMAPOATI	393
HSGPIPI1	1422	HSU43177	542	HUMBETAADB	40634
HSGSYG1	985	HSU43919	386	HUMBTFC	590
HSH4L	1317	HSU43920	291	HUMCW05	287
HSHCF1	17760	HSU46025	2898	HUMGCIA	3360
HSHFE	12146	HSU46165	5510	HUMHOXRAGE	10108
HSHFR3S02	4215	HSU48865	3706	HUMHSDII09	185
HSHIRAGN1	4018	HSU48869	2211		
HSHIRAHGN	1212	HSU49727	1689		
HSICAAR	3598	HSU49740	3149		
HSIGF2R1	1779	HSU49974	1301		
HSKETHEXK	637	HSU50061	278		
HSLEGUMAI	1393	HSU51039	720		
HSLWBGTPT	1946	HSU51241	1717		
HSMB1GENE	6502	HSU52427	6051		
HSMCP2	2991	HSU52852	7152		
HSMDCDIX1	1181	HSU56438	12177		
HSMEF2A10	4508	HSU56602	3480		
HSMEOPX10	411	HSU57316	2093		
HSMFH1	3289	HSU57655	626		

Tools used in this project

GRAILII	http://compbio.ornl.gov/Grail-1.3
FEXH	http://dot.imgen.bcm.tmc.edu:9331/gene-finder/gf.html
GENSCAN	http://CCR-081.mit.edu/GENSCAN.html
GenBank	http://www.ncbi.nlm.nih.gov
MATLAB neural network toolbox	