

Translation Initiation Sites Prediction with Mixture Gaussian Models

Guoliang Li¹, Tze-Yun Leong¹, Louxin Zhang²

¹ Medical Computing Laboratory, School of Computing

² Department of Mathematics

National University of Singapore

3 Science Drive 2, Singapore, 117543

{ligl, leongty}@comp.nus.edu.sg

matzlx@nus.edu.sg

Abstract. Translation initiation sites (TIS) are important signals in cDNA sequences. Many research efforts have tried to predict TIS in cDNA sequences. In this paper, we propose using mixture Gaussian models to predict TIS in cDNA sequences. Some new global measures are used to generate numerical features from cDNA sequences, such as the length of the open reading frame downstream from ATG, the number of other ATGs upstream and downstream from the current ATGs, etc. With these global features, the proposed method predicts TIS with sensitivity 98% and specificity 92%. The sensitivity is much better than that from other methods. We attribute the improvement in sensitivity to the nature of the global features and the mixture Gaussian models.

1 Introduction

Translation Initiation Sites (TIS) are the positions in cDNA sequences to start constructing proteins. The translation from cDNA to proteins, as we know, starts from TIS in a cDNA sequence and ends at the first in-frame stop codon downstream. It means if we know the TIS in one cDNA sequence, we will know the corresponding protein. Therefore, correct recognition of TIS can help us understand the gene structure, and its product.

Recognition of TIS in cDNA sequences is an important research topic that has been and is still being extensively examined [4,11-15]. In most cases, TIS is a trinucleotide ATG¹ (in DNA or cDNA) or AUG (in mRNA). However, there are numerous ATGs in cDNA sequences and only about one in thirty ATGs acts as TIS – this ATG is a functional ATG.

TIS is dependent on the position of ATG to the 5'-end of the cDNA sequences. As indicated in biological experiments, the first occurrence of codon ATG in a full-length, error-free cDNA sequence is a TIS in most of the known messenger RNA sequences. This inspired the scanning model hypothesis [4,12,15], which postulates that the small (40S) subunit of eukaryotic ribosomes initially binds at the 5'-end of mes-

¹ There are rare cases that other codons, such as ACG and CUG, are served as translation initiation sites. These will not be considered in this paper

senger RNA, migrates linearly downstream of the sequence, stops at the first AUG codon [15], and then translation process starts. Moreover, TIS is dependent on the context of the AUGs. Kozak first derived the consensus motif GCCRCCatgG around the TIS with statistical method in [11]. Within this motif, the purine in position² -3 and G in position +4 are the most highly conserved.

Although the scanning model hypothesis and consensus motif apply to most of the known messenger RNA sequences well, there are some notable exceptions [7,13,14] – the first ATGs are not TIS due to: 1) leaky scanning – the ribosome bypasses the first ATG codon – the putative start site – due to the very weak context, and translation starts from a downstream ATG with more optimal context; 2) reinitiation – translation starts from an ATG near the 5'-end of the messenger RNA and a small open reading frame (ORF) will be translated, but the ribosome continues scanning until the authentic ATG is reached to construct the protein; 3) internal initiation – the ribosome binds near the real ATG codon directly without scanning, which is reported for several viral mRNAs.

Advancement in technology has enabled more and more TIS to be verified by biological experiments. However, biological experiments are expensive and time-consuming. Therefore, computational methods are needed to help predict TIS in a cDNA sequence accurately and efficiently.

The consensus motif GCCRCCatgG around the TIS [11] was possibly the first attempt to identify TIS with statistical meaning. Although it is often used in biological experiments as a preliminary step to identify TIS in cDNA sequences, the motif is very rough and can't predict TIS well, since many fragments in cDNA sequences can match this consensus motif. Different data mining methods have been tried on TIS prediction problem, such as neural network [19], linear discriminant analysis [22], and support vector machine [26]. The common approach to solving the TIS prediction problem is to generate the numerical data from the cDNA sequences first, and then apply some computational methods to predict TIS.

To date, however, most of relevant features used in the existing prediction algorithms are local information. Little attempt has been made to generate numerical global features to predict TIS.

In this paper, we propose some measures to generate global features, and apply mixture Gaussian models to predict functional ATGs (which act as TIS) from all the occurrences of ATGs in cDNA sequences. With the global features, the proposed method can predict TIS with 98% sensitivity and 92% specificity, which represents a significant improvement in performance with respect to sensitivity.

2 Related works

Several past efforts focused on TIS prediction using different data mining methods. Stormo *et al* used neural network to identify TIS in *E. coli* [24], which is probably the first application of neural network technique to predict TIS. The other applications of neural network with comparable prediction performance can be found in [6,9,19]. The

² Numbering begins with the A of ATG as position +1 and increases downstream. The position just before ATG is numbered as -1 and decreases upstream.

numerical features used in these works are direct coding – which is a general way to generate numerical features from cDNA sequences: each nucleotide is encoded by four bits, 0001 for A, 0010 for C, 0100 for G, 1000 for T and 0000 for others. Coding difference between the region before and after TIS was used in [9].

Salamov *et al* developed the system *ATGpr* to identify TIS with linear discriminative analysis [22]. Six characteristics around ATG, such as the positional triplet weight matrix around an ATG and the ORF hexanucleotide characteristics were used to generate numerical data. Zien *et al* [26] engineered support vector machine to recognize TIS, probably with the best prediction performance so far. The measures they used to generate numerical features are as follows: direct coding, positional conditional matrix and other measures. The first-order Markovian dependencies and a dynamic program have been applied to the TIS prediction problem in [20,23], and the generalized second-order profiles were used in [1]. In addition to statistical information, Nishikawa *et al* [18] took the similarity of the cDNA sequences with protein sequences into consideration. The recent work by Nadershahi *et al* [17] compared several available computational methods for identifying TIS in EST data, and concluded that *ATGpr* is the best in the examined methods.

In the above efforts, the features used mainly take local information into account. As observed in [22,26], the data encoding measures affect the performance to recognize TIS from cDNA sequences. The experiments conducted by Pedersen and Nielsen [19] showed that relevant global information could improve prediction significantly.

3 Methods

In this work, we first propose some global measures to generate numerical data from the cDNA sequences. Then we apply the mixture Gaussian models to predict TIS from all occurrences of ATGs.

3.1 Proposed measures to generate numerical data

Many different data encoding measures can be used to generate numerical data from genomic sequences [8], including special data encoding measures for recognition of TIS, such as the consensus motif GCCACCatgG [11], the positional triplet weight matrix around an ATG, and the ORF hexanucleotide characteristics [22]. All of these features are local features, which take only the local information into account.

In the literature, some simple global features are also used, such as whether the ATG is the first ATG in the cDNA sequence. The first ATG is a strong feature to predict TIS from all occurrences of ATGs, which is supported by the scanning model hypothesis.

After examining the sequences, we observe that the length of the ORF in the cDNA sequences follows different distributions conditioned on the starting ATG – whether it is a TIS or not. From the histograms of the proposed features in the later section, this property is very clear. In another work (under preparation), we have compared several local features and drawn some preliminary conclusions as follows: 1) the direct coding measure generates too many features which makes each feature

less meaningful and 2) the higher-order position weight matrix and ORF hexanucleotide characteristics easily overfit the training data.

After careful consideration and comparison, the following measures are chosen for our experiments.

- 1) Length of upstream sequence from current ATG
- 2) Length of downstream sequence from current ATG
- 3) Log ratio of values in (2) / values in (1)
- 4) Number of upstream ATGs from current ATG
- 5) Number of downstream ATGs from current ATG
- 6) Log ratio of values in (5) / values in (4)
- 7) Number of inframe upstream ATGs from current ATG
- 8) Number of inframe downstream ATGs from current ATG
- 9) Log ratio of values in (8) / values in (7)
- 10) Number of upstream stop codons from current ATG
- 11) Number of downstream stop codons from current ATG
- 12) Log ratio of values in (11) / values in (10)
- 13) Number of inframe upstream stop codons from current ATG
- 14) Number of inframe downstream stop codons from current ATG
- 15) Log ratio of values in (14) / values in (13)
- 16) Length of open reading frame from current ATG

The numbers of ATGs and stop codons have been used in previous works, but the log ratio and the length of the open reading frame have not been used before. When we calculate the log ratio, we add pseudo count 1 to each value involved, since some values may be 0 under some cases. Note that some features in the list are functions of other features, e.g., feature 3 is the log ratio of feature 2 and feature 1. There are five such groups in total. We have done experiments to drop some features to avoid this type of functions. But, under all those cases, the performance of the system would deteriorate. A reasonable explanation is that the algorithm cannot learn the derived functions among the features properly.

3.2 Histograms of the numerical features

A histogram is simply a pictorial representation of a collection of observed data. It is particularly useful in forming a clear image of the true character of the data from a representative sample of the population. Usually the range of the attribute is divided into 7~15 bins and the frequencies of observations in each bin are counted and displayed in a graph. The graph will show how the observations distribute among the range. If the data follows a Gaussian distribution, the graph tends to cluster around an average and then taper away from this average on each side.

The histograms of the positive and negative training data are shown in Figure 1 and Figure 2 separately. Comparing these two figures, we can see that the distributions of positive and negative data of each feature are quite different. In particular, the distributions of all the log ratio features are near Gaussian. The means for the log ratio features for positive data are near 1, and the means for the log ratio features for negative data are around 0. It implies that the distributions of the features depend on the class labels.

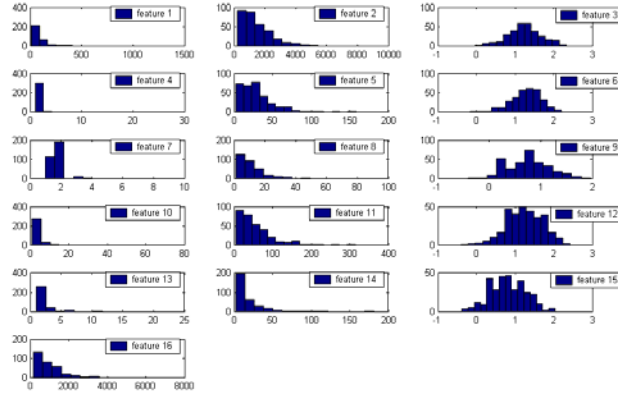


Fig. 1. Histograms of the positive training data – one for each feature

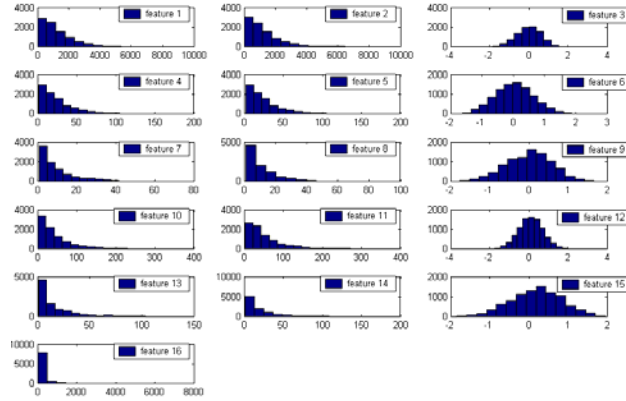


Fig. 2. Histograms of the negative training data – one for each feature

3.3 Mixture Gaussian models

The task of predicting TIS from all occurrences of ATGs in cDNA sequences is carried out as follows. Considering that the values of each feature are not randomly distributed, we make use of the histogram of each feature to distinguish TIS from all occurrences of ATGs. Referred to Figure 1 and 2, the peaks in the histograms of the positive data and negative data of each feature are quite different. The distributions of the positive data and the negative data of each feature can be approximated by mixture Gaussian models, since mixture models can approximate any continuous density to arbitrary accuracy provided the model has sufficiently large number of components and the parameters of the model are chosen correctly [2].

Mixture Gaussian model is a type of density model, which comprises a number of component Gaussian functions. Suppose the number of the components in the mixture Gaussian model is M . The class conditional density function of a data point \vec{x} belonging to class C is given by

$$p(\vec{x} | C) = \sum_{m=1}^M p(\vec{x} | m, C) p(m | C) \quad (1)$$

where $p(m | C)$ is the prior probability of the data point \vec{x} to be generated from component m of the mixture with probability $p(\vec{x} | m, C)$, which is a Gaussian as

$$p(\vec{x} | m, C) = (2\pi)^{-d/2} (\det \Sigma_m^C)^{-1/2} \exp\left\{-\frac{1}{2}(\vec{x} - \mu_m^C) \Sigma_m^{C-1} (\vec{x} - \mu_m^C)\right\}$$

where d is the dimension of the vector \vec{x} , μ_m^C is the mean vector of component m of class C and Σ_m^C is the covariance matrix of component m of class C . Here we assume that the covariance matrix of each Gaussian is some scalar multiple of the identity matrix, $\Sigma_m^C = (\delta_m^C)^2 \mathbf{I}$.

In the TIS prediction problem, the model is a two-class, two-component mixture model ($M=2$) – later section shows that $M=2$ is enough to model the data. Class 1 represents TIS. It is modeled by two 16 dimensional Gaussians (means and covariances) with associated mixing parameters. Class 2 represents the non-functional ATGs. It has a similar model as Class 1, but with different parameter values.

The structure of the model is shown in Figure 3 as a graphical model. Note that the square nodes represent discrete values and the round nodes represent continuous values. Node *class1/2* means that there are two classes, node *component1/2* with arrows from node *class* means that there are two components for each class, and node *Gaussian* means that the parameters μ_m^C and Σ_m^C depend on both the class and the component. Given the model, the parameters μ_m^C and Σ_m^C of the Gaussian mixture can be determined by the EM algorithm [5] with the training data belonging to that class.

In the E-step, the probability of each feature vector under different Gaussian components is calculated based on the existing parameters of the model. The recurrent equation is as follows.

$$p^{new}(m | \vec{x}, C) = \frac{p^{old}(m | C) p^{old}(\vec{x} | m, C)}{\sum_{m=1}^M p^{old}(m | C) p^{old}(\vec{x} | m, C)}$$

In the M-step, the parameters of the model are re-calculated as the sufficient statistics with the probabilities from the E-step.

$$p^{new}(m | C) = \frac{1}{N^C} \sum_{n=1}^{N^C} p^{new}(m | \vec{x}_n, C)$$

$$(\mu_m^C)^{new} = \frac{\sum_{n=1}^{N^C} p^{new}(m | \vec{x}_n, C) \vec{x}_n}{\sum_{n=1}^{N^C} p^{new}(m | \vec{x}_n, C)}$$

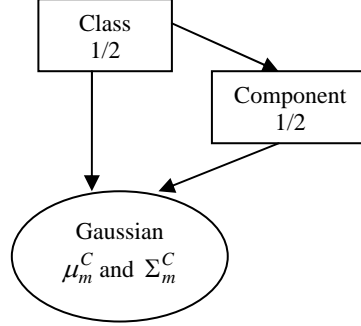


Fig. 3. The graphical representation of the mixture Gaussian model

$$((\sigma_m^C)^{new})^2 = \frac{1}{d} \frac{\sum_{n=1}^{N^C} p^{new}(m | \vec{x}_n, C) \left\| \vec{x}_n - (\mu_m^C)^{new} \right\|^2}{\sum_{n=1}^{N^C} p^{new}(m | \vec{x}_n, C)}$$

where N^C is the number of feature vectors belonging to class C .

With two Gaussian distributions for the representation of feature values of TIS and non-functional ATGs, we can get the probability of each feature vector generated by different classes with formula (1). Then the probability of each feature vector belonging to class C is

$$p(C | \vec{x}) = \frac{p(\vec{x} | C)p(C)}{\sum_C p(\vec{x} | C)p(C)}$$

Where $p(C)$ is the probability that a random feature vector belongs to class C , which is simply estimated from the training data.

In our study, there are only two classes. An optimal threshold to classify a feature vector is determined as the probability to generate the feature vector under Class 1 model which balances the sensitivity and specificity on the training data.

4 Experiments

To illustrate our observation we applied the mixture Gaussian models to a validated sequences set and compared the result with three other data mining methods. We also list the results from literature for reference.

4.1 Data set

Data is always a big issue in any data mining research. Here we choose the validated sequences set which has already been used successfully in [9] (personal communication with A. G. Hatzigeorgiou). The original sequence set is extracted from Swissprot. The steps are as follows: 1) collect human protein sequences whose N-terminal sites

are sequenced at the amino acid level (sequences manually checked by Amos Bai-roch); 2) retrieve the full-length mRNAs for these proteins whose TIS had been indirectly experimentally verified. 480 completely-sequenced and annotated human cDNAs were found; 3) divide the sequences set into training set and testing set before the experiments, 325 for training and 155 for testing.

After we got the original sequences, we generated numerical data with the proposed features. The feature vectors around TIS are positive data and those around the non-functional ATGs are negative data. There are 480 positives and 13628 negatives in total – the negatives are much more than the positives. This bias makes other classification methods difficult to recognize positives – many positives will be classified wrongly as negatives. This will be shown in the later section.

4.2 Evaluation measures

Prediction accuracy is measured by sensitivity and specificity. Let RP be the number of the total real positive ATGs in the data set, TP be the number of the total real positive ATGs predicted as positive, RN be the number of the total real negative ATGs in the data set, and TN be the number of the total real negative ATGs predicted as negative. Sensitivity (Se) is defined as TP/RP , and specificity (Sp) is defined as TN/RN .

4.3 Experiment Design

Generating the numerical data is the first important step of the experiment. The sequences and the encoding measures have been mentioned above. The total sequences are split into two sets: 325 for training and 155 for testing. There are 325 positive ATGs and 9489 negative ATGs in training set, and there are 155 positive ATGs and 4139 negative ATGs.

The mixture Gaussian model is built in Matlab with support of the Bayes Net Toolbox (BNT) [16]. The BNT is a special software package for manipulating graphical models and Bayesian networks. It supports most of the important methods (inference, parameter learning, and structure learning) for graphical models. This makes training mixture Gaussian models easier.

In our model, it contains three nodes: one for different classes, one for different components and one for all the Gaussian vectors. The parameters are learned by the EM algorithm. The EM algorithm works by starting with randomly initialized parameters, and then iteratively refines the model parameters to produce a locally optimal maximum-likelihood fit or stops when the number of the maximum iterations reaches. The number of the maximum iterations is determined by experiments. It is set to 10 in our experiments.

After the trained model is available, it is used to classify the training samples to determine the threshold. We adopt such a strategy to choose a threshold: when such a threshold is chosen, the sensitivity and specificity on the training set should be balanced – the sensitivity is equal to or approximately equal to the specificity. The value of the threshold is model-dependent, since each model is trained with random initial

parameters. However, the performance on the testing data is similar when the testing data is tested with the same model.

Another factor that affects the performance of the system is the number of components in the model. We have tested with 1, 2, 3, 4, 5, 6, 8, 10 components separately. The classification result on the testing data is shown in the following Figure 4.

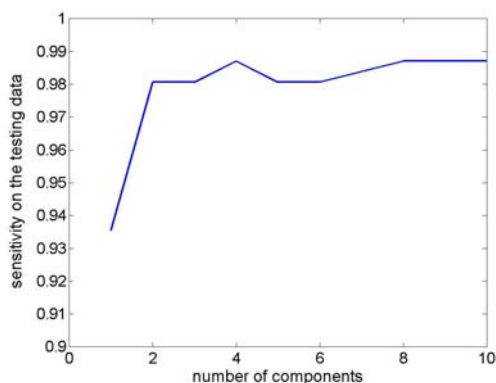


Fig. 4. The relationship of the number of components and the sensitivity of the system on the testing data

From Figure 4, we know that there is an increase in sensitivity when the Gaussian model changes from one- to two-components. After that, the sensitivity remains around a stable value. Therefore we chose 2 as the number of components in our experiments.

4.4 Comparison with other methods

Three other data mining methods – decision tree, support vector machine, and logistic regression – are adopted to build classification model with the same data to build mixture Gaussian model for comparison purpose. Decision tree method [21] is a *de facto* classification method to evaluate other classification method. Support vector machine [3] is possibly the classification method which have the best prediction result up to date, although it sometimes suffers from noisy data. Logistic regression [10] is a non-linear transformation of the linear regression, which applies to the case when the dependent variable is discrete. A well-known machine learning package Weka [25] has these three methods implemented. We ran these three methods with the default parameters on the same training and test data used in mixture Gaussian model. The results are shown in Table 1.

From Table 1, we can see the mixture Gaussian model can predict TIS with very high sensitivity, and the system performs with balanced sensitivity and specificity. The other three methods are better in terms of specificity of the prediction results, but their sensitivities are quite lower. It means that many TIS are missed in the prediction – this makes the prediction less meaningful.

The best results in the literature to predict TIS are listed in Table 2. In these works, the sequences used to generate these results are ESTs, not full-length cDNA se-

quences. The global features defined in this work are not considered in these works. Hence, we cannot run our method on their sequences. Consequently, we also cannot at this point conclude that our method is better than all or most of the existing methods, although in terms of sensitivity and specificity our numbers are “better”. These results are listed here for reference only.

Method	Sensitivity (%)	Specificity (%)
Mixture Gaussian model	98.06	92.41
Decision tree	80	99.68
Support vector machine	67.74	99.37
Logistic regression	76.77	99.46

Table 1. The comparison of the results from 4 different methods – mixture Gaussian models, decision tree, support vector machine and logistic regression

Method	Sensitivity (%)	Specificity (%)
Neural network ¹	82.4	64.5
Salzberg method ²	68.1	73.7
SVM ³	78.4	76

Table 2. The results from literature. Note: The data for these three methods are from [26].

1) The original work for neural network is in [19]. 2) The original work for Salzberg method is in [23]. 3) The original work for SVM is in [26].

5 Conclusion

In our system, we have proposed new measures to generate global features and applied mixture Gaussian models for predicting TIS in cDNA sequences. The numbers of ATGs and stop codons around ATG have been used in literature before. Other features, such as the log ratio of the lengths of down stream sequences and upstream sequences from ATGs, the log ratio of the numbers of down stream ATGs and stop codons, the length of the open reading frame down stream the ATGs, are used for the first time to predict functional ATGs from non-functional ATGs. From the histograms of these features, we can observe that these features follow the Gaussian distribution or approximate Gaussian distribution. The mixture Gaussian models are natural and efficient to model these phenomena.

Our mixture Gaussian model is trained with the EM algorithm. When the trained model is applied on the TIS prediction problem, it performs much better than other methods in terms of sensitivity. This means that the proposed global features and mixture Gaussian models are good for the TIS prediction problem. Two specific features and their related features should be mentioned: One is the number of upstream ATGs, which contains the information of whether one ATG is the first ATG in the cDNA sequence and coincides with the scanning model hypothesis. The other is the number of

downstream stop codons, which contains the information of whether there is a stop codon downstream – important information about the completeness of the ORF.

A possible problem in the proposed method is that it requires full-length cDNA sequences to generate global features. Since it is getting easier to get full-length cDNA sequences and more full-length cDNA sequences are available now, this problem will be alleviated in the future.

Acknowledgments

We are grateful to Artemis Hatzigeorgiou, who kindly provided the original cDNA sequences. This research is supported by Research Grant No. R-252-000-111-112/303 from the Biomedical Research Council (BMRC) of the Agency for Science, Technology, and Research (A*Star) and the Ministry of Education in Singapore. L. Zhang was supported by BMRC Research Grant BMRC01/1/21/19/140.

References

- [1] P.K. Agarwal, V. Bafna, Detecting non-adjointing correlations within signals in DNA, in: Proceeding of the 2nd Annual International Conference on Computational Molecular Biology RECOMB (1998) 2-8.
- [2] C.M. Bishop, Neural networks for pattern recognition (Clarendon Press, Oxford, 1995).
- [3] C.J.C. Burges, A Tutorial on Support Vector Machines for Pattern Recognition, *Data Mining and Knowledge Discovery* 2 (1998) 121-167.
- [4] A. Cigan, L. Feng, T. Donahue, tRNAⁱ(met) functions in directing the scanning ribosome to the start site of translation, *Science* 242 (1988) 93-97.
- [5] A.P. Dempster, N.M. Laird, D.B. Rubin, Maximum Likelihood from Incomplete Data via The EM Algorithm, *Journal of Royal Statistical Society* 39 (1977) 1-38.
- [6] C. Derst, M. Reczko, A. Hatzigeorgiou, Prediction of human translational initiation sites using a multiple neural network approach, *The International Journal of Computers, Systems and Signals* 1 (2000) 169-179.
- [7] T.E. Dever, Gene-specific regulation by general translation factors, *Cell* 108 (2002) 545-556.
- [8] J.W. Fickett, The gene identification problem: an overview for developers, *Computer & Chemistry* 20 (1996) 103-108.
- [9] A.G. Hatzigeorgiou, Translation initiation start prediction in human cDNAs with high accuracy, *Bioinformatics* 18 (2002) 343-350.
- [10] D.W. Hosmer, S. Lemeshow, *Applied logistic regression* (John Wiley & Sons, New York, 2000).
- [11] M. Kozak, At least six nucleotides preceding the AUG initiator codon enhance translation in mammalian cells, *Molecular Biology* 196 (1987) 947-950.
- [12] M. Kozak, How do eucaryotic ribosomes select initiation regions in messenger RNA?, *Cell* 15 (1978) 1109-1123.

- [13] M. Kozak, Interpreting cDNA sequences: some insights from studies on translation, *Mammalian Genome* 7 (1996).
- [14] M. Kozak, Pushing the limits of the scanning mechanism for initiation of translation, *Gene* 299 (2002).
- [15] M. Kozak, The scanning model for translation: an update, *Cell Biology* 108 (1989) 229-241.
- [16] K. Murphy, Bayes Net Toolbox for Matlab, <http://www.ai.mit.edu/~murphyk/Software/BNT/bnt.html>, in: (2004).
- [17] A. Nadershahi, S.C. Fahrenkrug, L.B.M. Ellis, Comparison of computational methods for identifying translation initiation sites in EST data, *BMC Bioinformatics* 5 (2004).
- [18] T. Nishikawa, T. Ota, T. Isogai, Prediction whether a human cDNA sequence contains initiation codon by combining statistical information and similarity with protein sequences., *Bioinformatics* 16 (2000) 960-967.
- [19] A. Pedersen, H. Nielsen, Neural network prediction of translation initiation sites in eukaryotes: perspectives for EST and genome analysis, in: T. Gaasterland, P.D. Karp, K. Karplus, C.A. Ouzounis, C. Sander, A. Valencia (Eds.), *Proceedings of the 5th International Conference on Intelligent Systems for Molecular Biology ISMB'97* (AAAI Press, Halkidiki, Greece, 1997) 226-233.
- [20] M. Pertea, S. Salzberg, A Method to Improve the Performance of Translation Start Site Detection and Its Application for Gene Finding, in: *Proceeding of the 2nd Workshop on Algorithms in Bioinformatics (WABI2002)* (2002) 210-219.
- [21] J.R. Quinlan, *C4.5: programs for machine learning* (Morgan Kaufmann, San Mateo, Calif., 1993).
- [22] A. Salamov, T. Nishikawa, M.B. Swindells, Assessing protein coding region integrity in cDNA sequencing projects, *Bioinformatics* 14 (1998) 384-390.
- [23] S. Salzberg, A method for identifying splice sites and translational start sites in eukaryotic mRNA, *Computer Applications in Biosciences (CABIOS)* 13 (1997) 365-376.
- [24] G.D. Stormo, T.D. Schneider, L. Gold, A. Ehrenfeucht, Use of the 'Perceptron' algorithm to distinguish translational initiation sites in *E. coli.*, *Nucleic Acids Res* 10 (1982).
- [25] I.H. Witten, E. Frank, *Data mining: practical machine learning tools and techniques with Java implementations* (Morgan Kaufmann, San Francisco, 1999).
- [26] A. Zien, G. Ratsch, S. Mika, B. Scholkopf, T. Lengauer, K.-R. Muller, Engineering support vector machine kernels that recognize translation initiation sites, *Bioinformatics* 16 (2000) 799-807.