BIOINFORMATICS

Vol. 17 no. 10 2001
Pages 913–919

DIANA-EST: a statistical analysis

Artemis G. Hatzigeorgiou[1, 2], Petko Fiziev[1] and Martin Reczko[2]

[1]Metagen GmbH, Ihnestr.63, 14195 Berlin, Germany and [2]Synaptic Ltd, Science and Technology Park of Crete, PO Box 1447, Voutes Heraklion, 71110 Greece

Received on April 20, 2001; revised and accepted on July 12, 2001

ABSTRACT

Motivation: Expressed Sequence Tags (ESTs) are next to cDNA sequences as the most direct way to locate *in silico* the genes of the genome and determine their structure. Currently ESTs make up more than 60% of all the database entries. The goal of this work is the development of a new program called DNA Intelligent Analysis for ESTs (DIANA-EST) based on a combination of Artificial Neural Networks (ANN) and statistics for the characterization of the coding regions within ESTs and the reconstruction of the encoded protein.

Results: 89.7% of the nucleotides from an independent test set with 127 ESTs were predicted correctly as to whether they are coding or non coding.

Availability: The program is available upon request from the author.

Contact: Present address: Department of Genetics, University of Pennsylvania, School of Medicine, 475 Clinical Research Building, 415 Curie Boulevard, Philadelphia, PA 19104-6145, USA. artemis@pcbi.upenn.edu.

INTRODUCTION

The most direct way to characterize the coding regions of genomes and provide reliable information for structural annotation of genes in genomic sequences still remains the analysis of sequences from cDNA libraries. As a consequence of their contribution to rapid gene discovery, full and partial cDNA sequences have been generated in very large numbers, both in public and private sectors. Expressed Sequence Tags (ESTs) make up currently more than 60% of all the database entries and EST sequencing projects have already started to have a major impact on biomedical research, by accelerating the identification of new genes of interest as potential targets for drug discovery, and by providing target sequences for genome wide expression profiling.

Unlike high quality finished genome sequences, which are double-strand and multiple-pass, cDNA and EST sequences are mostly single-strand, single-pass sequences which contain sequencing errors. Errors may result in nucleotide substitutions, insertions or deletions, leading to frame-shifts while analyzing these sequences (States and Botstein, 1991; Posfai and Roberts, 1992; Claverie, 1993). The analysis of ESTs is further complicated by the fact that they are usually 300–600 nucleotides long, originate from different parts of the cDNA, and may include only sequences of non-translated regions.

Although the number of novel cDNA and EST sequences is growing very rapidly few programs (apart from clustering algorithms) have been developed for their annotation.

The most common way to find frame-shift errors in coding sequences is the use of similarity searches between target sequences and other known sequences on the DNA and protein level (States and Botstein, 1991; Posfai and Roberts, 1992; Guan and Uberbacher, 1996; Birney *et al.*, 1996). However there also exist a small number of programs which identify sequencing errors based only on statistic analysis (Fichant and Quentin, 1995; Xu *et al.*, 1995). Xu *et al.* (1995) used hexamer in frame occurrences to predict the coding frame and dynamic programming to locate the exact position of the errors. The algorithm was mainly designed to work on genomic sequences and in combination with gene prediction programs. Fichant and Quentin (1995) used the frequency of dicodons in the DNA sequence in combination with correspondence analysis for the prediction of the coding frame and a selection of heuristic rules for the exact location of the error position.

One of the programs specially designed for finding the coding regions in ESTs is ESTScan (Iseli *et al.*, 1999), which uses a new type of Hidden Markov Model (HMM) and can deal with the possibility of errors in the analyzed sequence. However, this program does not include a feature for special recognition of the start and the end of the coding region and can therefore miss the exact location of a start or a stop codon.

The goal of this work is to investigate a new method for the analysis of EST nucleotide sequences. The programs makes use of the experimentally verified data for full-length cDNA and corresponding EST sequences and provides a holistic set of compact and efficient tools that can help in different steps of the nucleotide sequence analysis. In addition to the frame-shift tolerant coding

© Oxford University Press 2001

region recognition, this program also performs recognition of the start codon (or Translation Initiation Site, TIS) and stop codon at the end of the coding region.

The algorithm is based on a combination of ANN and statistics. An ANN is basically a computer program that detects patterns and correlations in data. In applications with regard to nucleotide or amino acid analysis, it can learn to recognize and classify a sequence pattern by increasing the emphasis placed on important information and ignoring irrelevant information. ANN-training incorporates both positive and negative information, that means in this case DNA sequences with and without the feature of interest. Furthermore, ANNs are able to detect second- and higher-order correlations in patterns, an approach that can find more complex correlations than a method based simply on the frequency of occurrence of nucleotides at certain positions. A preconceived model is not required, because ANN automatically determines which residues and which positions are important (Hirst and Sternberg, 1992).

MATERIALS

In order to obtain a validated data set, the first step of data collection was made using the protein database Swissprot, rather than a genomic database. All the human proteins whose starts were sequenced at the amino acid level were manually collected (by Amos Bairoch, personal communication). The next step was to retrieve the full-length mRNAs for these proteins for which the TIS was now indirectly experimentally verified. 475 corresponding human cDNAs, completely sequenced and annotated, were found. The average size of the cDNAs length was 1617 nucleotides. Out of these 475 cDNAs, three-quarters were used for the extraction of the training data, here called the training gene-pool, One quarter was used for extraction of the test data, here called the test gene-pool. For the full design of the modular structure of the algorithm, several datasets were constructed. These datasets were used for:

- the training of the ANNs, called the ANN-training set,
- the evaluation of the generalization performance of the ANNs during the training, called the ANN-evaluation set and,
- the final testing of the performance of a trained ANN, called the ANN-test set.

ANN-training and evaluation sets were extracted from the training gene pool, while the ANN-test set was extracted from the test gene pool. In the last step of the algorithm, different modules were integrated into one approach.

In the second step a search through the EMBL database was made for EST entries corresponding to these cDNAs. The search results gave 242 ESTs for the training dataset and 127 for the test dataset. The average length of the selected ESTs was 592 nucleotides. Through the alignment of the corresponding ESTs and cDNAs it was possible to determine for each EST two characteristics, (a) if it was identifed as a non-coding or as coding region, and (b) on which strand it was coding and whether it contained start or stop codons. This dataset was used for training and determining the parameters of the EST analysis algorithm.

Consensus-ANN

For the consensus-ANN a window 12 nucleotides long was used. This sequence includes the positions from -7 to $+5$, where $+1$ is the position of the first nucleotide of an ATG triplet. Each cDNA sequence provides only one positive data example for TIS. Consequently, only a relatively small amount of data was used to train this ANN (325 positive and 325 negative examples). The negative examples were collected from the UTR and coding regions. All negative examples had an ATG at the 8th position of the selected sequence-window. The input is presented to the network through the universal encoding system, where each nucleotide is transformed into a binary 4-digit string (Figure 1). A number of different feed forward ANN architectures were tested during the training procedure:

- feed forward nets without hidden units (perceptron),
- feed forward nets with hidden units, and
- feed forward nets with hidden units and short cut connections (direct connections from the input to the output units).

In the second and third case, a number of different hidden units were tested (results not shown). The ANN with the highest score according to the correlation coefficient measure was chosen. This was the ANN with short cut connections and two hidden units.

In this case, the training was performed with cascade correlation (Fahlman and Lebiere, 1990). In cascade correlation, training starts with a perceptron, which is an ANN with weights only between the input and the output units. After a number of iterations, a group of the weights freeze (stop changing) and a hidden unit is added. In the remaining iteration, the new weights are trained to learn examples which had not been successfully taught in the first steps of the training.

The performance of TIS prediction on the test set has an accuracy of 76.4%, where the accuracy is taken to be the average of the prediction on the positive and the negative data.

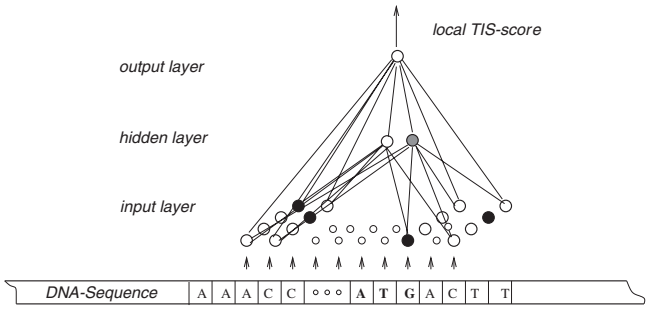

Fig. 1. The architecture of the module for the recognition of the consensus motif around TIS.

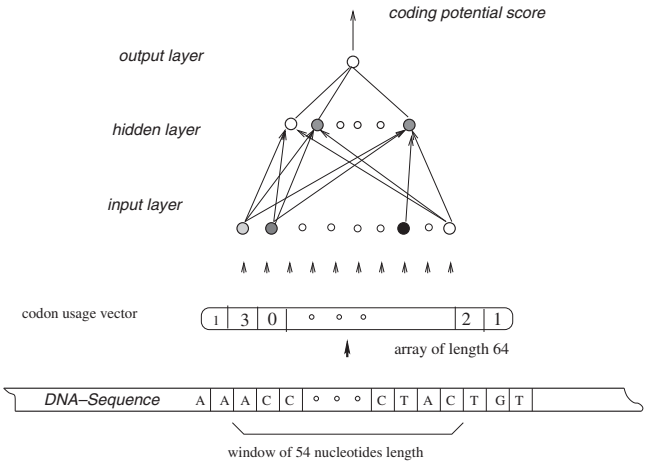

Fig. 2. The architecture of the module for the recognition of the coding region around TIS. A sequence window of 54 nucleotides length is transformed into a 64 long vector, which is then used as an input to the ANN (refer to text).

Coding-ANN

In the second step, an ANN was trained for the recognition of the coding region. In this case the window size of the sequences used was 54 nucleotides in length. For the training of this second ANN it was possible to extract multiple positive data from every gene. Positive examples were extracted from the coding region and always started with the first nucleotide of a codon (in frame). Negative examples were extracted from the non-coding regions randomly and from those windows out of the coding region, which start with the second and third nucleotide of a codon (window out of frame). Here, a possible homology between training and test data could influence the result. For this reason, such homologies between the training and test genes were eliminated through pair-wise alignment with the full Smith–Waterman algorithm. Only the genes from the training pool with less than 70% homology to the genes of the test pool were used for extracting the training data—a total of 282 genes. From these genes, 700 positive and 700 negative sequence regions were extracted. An additional 500 regions (50% positive–50% negative) were extracted from the test gene pool for testing the performance of the coding-ANN.

Previous investigation has shown that preprocessing the data through a coding measure can significantly improve the performance of the ANN (Hatzigeorgiou *et al.*, 1999). A variety of coding measure methods exists. The best results are obtained by applying the codon usage statistic to the sequence window. With other words the number of all different codons (64) on the sequences window of 54 nucleotides were calculated. The counting starts with the first nucleotide of the window, counting all non-overlapping codons.

This results in a transformation of the sequence window to a vector of 64 units. Each unit gives the frequency (normalized) of the corresponding codon appearing in the window (Figure 2) .

If the window starts with the first nucleotide of a codon, the ANN has a high score (close to 1), otherwise the score is low (close to 0).

The training is done with the algorithm Resilient Backpropagation (RPROP; Riedmiller and Braun, 1993, an improved version of the *classical* Backpropagation), applied to a feed forward ANN.

The main differences between RPROP and Backpropagation are:

- the change of the weights depends only on the sign of the potential derivation of the error and not on its size;
- the weight update phase incorporates the current gradient and the gradient of the previous step; and
- every weight has its own learning parameter for the changing of the weight value.

The experiments show that RPROP yields to an improvement in generalization performance of around 2% over the Backpropagation algorithm. A neural network with two hidden units was experimentally defined to give the best performance (Hatzigeorgiou *et al.*, 1999).

The performance of coding-ANN on the test set has an accuracy of 84%, where the accuracy is taken to be the average of the prediction on the positive and the negative data.

Frame-ANN

For the development of a frame sensitive module an ANN is trained with the same procedure as described in the Section **Coding-ANN**. In the same way, the sequence window was transformed into a 64 vector units, which then

was used as an input to the ANN. The only difference between the two procedures is in the composition of the training data. While the positive dataset is the same (windows extracted from the coding region in frame), the negative data are in this case only extracted out of the coding region starting with the second and third nucleotide of a codon.

The prediction accuracy of the frame-ANN increases from 84 to 87.5% in comparison with the coding-ANN. The better prediction leads to a better estimation of the transaction between two frames.

Generalization of the ANNs

One of the critical issues in the training of an ANN is determining the appropriate moment to stop training. If the training of the ANN is long, this can decrease the global error of the training set, but on the other hand may also lead to over-training the ANN. In the latter case, the ANN is taught the characteristic of individual examples and not their global characteristics.

In order to avoid this, only 2/3 of the examples of the training set are used for training; the remaining 1/3 are used for evaluating the performance of the ANN following every iteration. Once the performance of the *evaluation* group of examples starts to decrease, the ANN is stopped. As mentioned in the dataset section, the extraction of the evaluation examples is made from genes belonging to the training pool rather than from the test pool.

All the training of the ANNs is performed by the Stuttgarter Neural Network Simulator (SNNS), publicly available from the University of Stuttgart, Germany (Zell *et al.*, 1993).

INTEGRATED METHOD

For the prediction of the coding regions in ESTs all three modules are integrated in one approach.

In the first step the frame-ANN is applied on a sliding window along the sequence. If the sequence is derived from a coding region without sequencing errors the output will be a number series with a high score in every third position. This is the position of the first nucleotide of a codon. If a deletion or insertion occurs, this periodicity will get disturbed. In the ideal case the numeric chain of a nucleotide sequence will be the alternation of 100, starting with 1 (for example: $100100100 \cdots 100100$). In the second step of the algorithm the ideal chain of the sequence gets aligned—using a dynamic programming approach—with the real score values in order to maximize the overall coding score potential (which, in this case, is calculated by multiplying the values of the two sequences). Finally, retracing back the path of the best alignment (the alignment with the best score) it is possible to locate the frame changes. This method is similar with the approach described in Xu *et al.* (1995). In order to avoid a frequent frame-switching it is necessary to introduce a frame-change penalty. Only frame-changes longer than 60 nucleotides are allowed. In other words a new frame change is allowed only if this new frame is active for more than 60 nucleotides. However, it is possible for the user to change the frame-switching penalty and experiment with different parameters. A higher penalty will give a more accurate coding region prediction and a better strand prediction. A lower penalty will give a more accurate reconstruction of the protein sequence encoded in an EST.

In the following step the coding potential along the sequence is calculated by using the information of the leading frame in every region. More precisely the coding score of the sequence is calculated by adding the coding score of every window in the leading frame. If the sequence gives a high coding score the search continues for the determination of the start and/or the stop codon. For the recognition of the coding start the consensus-ANN is used to calculate a score for every ATG (putative TIS). In addition the non-coding/coding potential around every ATG is calculated by building the difference between all coding scores (calculated by the coding ANN) of in-frame 60 positions before and after the ATG. If the product of the consensus score and the non-coding/coding difference is above a certain threshold (here 0.2) and the ATG is on the leading frame, the ATG is characterized as TIS.

Stop codons are permitted on the predicted coding region. Possible ends of the coding sequence are determined by the presence of stop codons in a local coding frame. Local coding frame means that the frame of the stop-codon has a high score for 60 nucleotides before the stop codon.

The output of the program gives the start and the end of the coding region, the coding score of the coding region, the score for the predicted TIS and the strand on which the coding region is determined. For every subsequence that is characterized by continuous coding frame, the coding potential and the translated amino acid sequence is given. At the end of the output the whole sequence of the suggested protein is printed.

RESULTS AND DISCUSSION

The program was evaluated with a set of 127 ESTs. 109 sequences contained coding regions. The combination of all the above methods gives an accuracy of 89.7% for prediction of both coding and non-coding nucleotides in the EST test dataset, containing a total of 146 648 nucleotides.

For 101 out of the 109 sequences that contain coding regions the program was able to detect the correct coding strand. Out of the 18 sequences with no coding regions was able to detect 7 as non-coding at all. For 77 out of the 96 sequences that were predicted as coding, the program was able to detect the exact start of the coding region. This

could be a translation initiation start-site or just the start of the sequence. For 29 sequences out of the 127 the program was able to detect the correct end of the coding region.

Table 1 presents the predicted coding regions of the program for a subset of the test-set sequences. The third and fifth columns indicate the correct and the predicted strand, the fourth column the true start and end of the coding regions and the sixth column the predicted start and end of the coding region. The second column gives the length of the EST.

A graphical representation of the prediction along an EST with a frame-shift around the position 270 is given in the last figures. Figure 5 illustrates the graphical output of the program for prediction of TIS, Figure 3 shows the prediction along the sequence with the frame-ANN and Figure 4 shows the prediction with the coding-ANN.

For practical use of the program the output contains not only the suggested amino acid sequence (including the automatic correction of the frame-shifts), but also a quality prediction for every subpart of the sequence. This makes it possible for the user to start functional analysis from the amino acid sequences which are predicted with the highest score and are most likely to be correct.

On a comparison with ESTscan a set of 107 ESTs was used. The sensitivity (cp/cp + fn) for ESTScan was 88.7% and of DIANA-EST 86.5%. The specificity (cp/cp + fp) of the prediction was for ESTScan 87.3% and of DIANA-EST 79.6%. The accurate prediction of the start of the coding region (plus order minus 10 nucleotides) was correct for 37 sequences predicted by ESTScan and 76 predicted by DIANA-EST. The correct prediction of the end of the coding region was predicted in 43 cases correct from ESTscan and for 56 sequences correct from DIANA-EST.

The software we are presenting here is a useful additional tool for the functional analysis of ESTs. In functional analysis of ESTs it is common that domain predictions are made on the six frame translations of the sequences. This leads to a very high number of false positive hits. Additionally it is possible that domains located at a frame transition (include a frame-shift) will not be predicted through functional analysis tools. Using the program described here it is possible to run the search only on predicted and frame-shift corrected protein sequences.

Although all the training and test sets of the method presented here is derived from human sequences it is possible to retrain the same method for the analysis for ESTs derived from other species. Furthermore, we believe this program to be a good addition for the analysis of ESTs with ESTScan. The fact that the two programs are based on different mathematical models (HMM for ESTScan and ANN for DIANA-EST) makes their use complementary.

This work is currently in progress and there is poten-

Table 1. The predicted coding regions of the program for a subset of the testset sequences

Analytical results on a subset of the testset										
		True coding region				Predicted coding region				
Name	Length	Strand	Start	–	End	Strand	Start	–	End	
AA393215	503	0	0	–	65	0	1	–	344	
AA574223	550	0	0	–	550	0	86	–	550	
AA203389	898	0	0	–	536	0	1	–	530	
AA634117	833	1	0	–	0	1	164	–	511	
AA570193	450	1	38	–	334	0	1	–	421	
AA477734	577	0	51	–	185	0	1	–	542	
AA669154	845	1	0	–	0	0	1	–	605	
AA196380	661	1	38	–	520	1	38	–	519	
AA194299	613	1	0	–	439	0	84	–	613	
AA554510	697	1	33	–	560	1	1	–	555	
AA642200	836	1	119	–	670	1	119	–	670	
F20811	378	0	49	–	234	0	49	–	234	
AA629017	548	1	0	–	292	1	1	–	290	
AA742536	348	1	47	–	202	1	46	–	235	
AA383080	324	0	0	–	73	0	0	–	0	
AA205573	666	1	0	–	372	1	55	–	372	
W73203	401	0	57	–	242	0	56	–	268	
AA126791	686	1	0	–	485	0	56	–	434	
W79564	390	0	36	–	317	0	35	–	316	
AA583193	626	1	0	–	252	1	46	–	252	
AA557174	525	1	61	–	324	1	61	–	324	
AA706930	472	1	26	–	409	1	1	–	436	
AA059484	628	0	0	–	0	0	0	–	0	
W80489	537	1	0	–	319	0	258	–	537	
H47306	500	0	30	–	500	0	30	–	500	
AA551088	707	1	42	–	464	1	1	–	458	
AA580413	537	1	0	–	312	1	1	–	309	
AA460603	578	0	0	–	578	0	1	–	578	
AA779742	637	1	0	–	297	0	286	–	606	
W84553	595	1	61	–	354	0	1	–	468	
AA702482	611	1	0	–	213	1	1	–	611	
AA576597	598	1	0	–	527	1	1	–	598	
AA554746	719	1	0	–	256	1	1	–	192	
AA505266	584	1	0	–	507	1	1	–	499	
AA432244	602	1	0	–	410	1	1	–	492	
AA305389	494	0	0	–	494	0	1	–	494	
AA442263	521	1	47	–	450	1	47	–	455	
AA779820	599	1	0	–	383	0	1	–	599	
AA694189	672	1	0	–	435	1	1	–	403	
AA587388	666	1	0	–	666	1	1	–	625	
AA576463	542	1	0	–	146	0	1	–	542	
AA155944	592	0	90	–	407	0	90	–	407	
AA282673	538	0	0	–	538	0	1	–	415	
AA600818	465	1	81	–	215	1	81	–	215	
AA280261	617	0	76	–	459	0	74	–	617	
AA642123	951	1	233	–	751	1	256	–	743	
AA535872	590	0	57	–	404	0	57	–	403	
AA573848	622	1	0	–	523	1	1	–	622	
AA625520	527	0	111	–	410	0	109	–	408	
AA037480	509	1	0	–	0	0	1	–	463	
AA430281	581	0	96	–	554	0	96	–	554	
AA602855	589	1	0	–	304	1	1	–	305	
AA126722	565	1	0	–	469	1	1	–	473	
AA034144	703	0	0	–	703	0	1	–	429	
W44675	588	0	0	–	588	0	1	–	588	
AA005020	631	0	0	–	0	0	29	–	336	
AA572843	631	0	0	–	276	0	1	–	631	
AA514315	593	1	0	–	300	1	1	–	299	

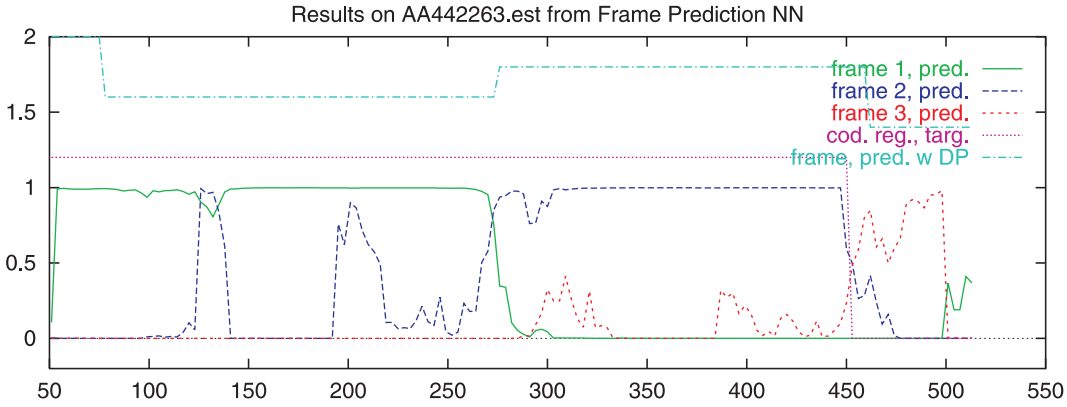

Fig. 3. The frame based prediction along an EST. The real start is at the position 45. The line between the position 50 and 450 indicates the real coding region of the EST. The three different frames and the prediction through dynamical programming are shown. The score is 0 for a not active frame and 1 for an active frame.

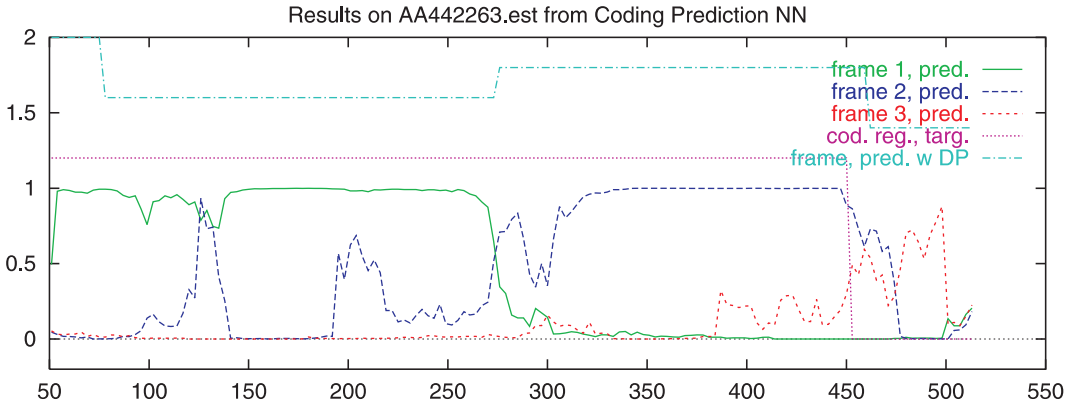

Fig. 4. The coding-based prediction along an EST. The real start is at the position 45. The line between the position 50 and 450 indicates the real coding region of the EST. The three different frames and the prediction through dynamical programming are shown. The score is 0 in a non-coding region and 1 in a coding region.

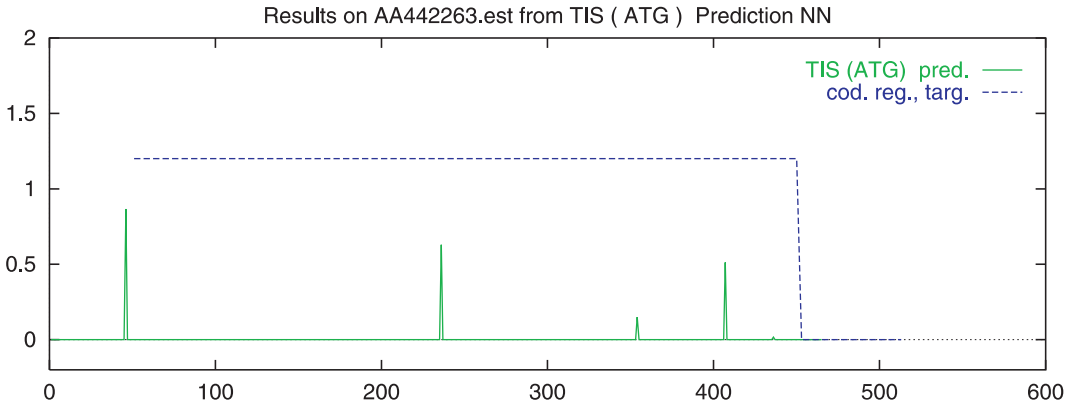

Fig. 5. The prediction of TIS along an EST. The real start is at the position 45. The line between the position 50 and 450 indicates the real coding region of the EST.

tial for improving the results by different steps of the algorithm, like cooperating similarity search with protein databases.

ACKNOWLEDGEMENTS

We are thankful to James W.Fickett for helpful comments, to Amos Bairoch for providing the dataset and to the anonymous reviewers for helpful comments. This work was supported by the Greek Secretary of Research & Development.

REFERENCES

Birney,E., Thompson,J.D. and Gibson,T.J. (1996) Pairwise and searchwise: finding the optimal alignment in a simultaneous comparison of a protein profile against all DNA translation frames. *Nucleic Acids Res.*, **24**, 2730–2739.

Claverie,J-M. (1993) Detecting frame shifts by amino acid sequence comparison. *J. Mol. Biol.*, **234**, 1140–1157.

Fahlman,S.E. and Lebiere,C. (1990) The cascade-correlation learning architecture. In Touretzky,D.S. (ed.), *Advances in Neural Information Processing systems II*. Morgan Kaufmann, Los Altos, CA, pp. 524–532.

Fichant,G.A. and Quentin,Y. (1995) A frameshift error detection algorithm for DNA sequencing projects. *Nucleic Acids Res.*, **23**, 2900–2908.

Guan,X. and Uberbacher,E.C. (1996) Alignments of DNA and protein sequences containing frameshift errors. *Comput. Appl. Biosci.*, **12**, 31–40.

Hatzigeorgiou,A.G., Papanikolaou,H. and Reczko,M. (1999) Finding the reading frame in protein coding regions on DNA sequences: a combination of statistical and neural network methods. In Mohammadian,M. (ed.), *Computational Intelligence: Neural Networks & Advanced Control Strategies*. IOS Press, Vienna, pp. 148–158.

Hatzigeorgiou,A. and Reczko,M. (1999) Feature recognition on expressed sequence tags in human DNA. In *Proceedings of the International Joint Conference on Neural Networks*. CD, INNS Press.

Hirst,J.D. and Sternberg,M.J. (1992) Prediction of structural and functional features of protein and nucleic acid sequences by artificial neural network. *Biochemistry*, **31**, 7211–7218.

Iseli,C., Jongeneel,C.V. and Bucher,P. (1999) Estscan: a program for detecting, evaluating and reconstructing potential coding regions in EST sequences. In *Proceedings of the Intelligent Systems for Molecular Biology*. AAAI Press, Menlo Park, CA.

Posfai,J. and Roberts,R.J. (1992) Finding errors in DNA sequences. *Proc. Natl Acad. Sci. USA*, **89**, 4698–4702.

Riedmiller,M. and Braun,H. (1993) A direct adaptive method for faster backpropagation learning: the RPROP algorithm. In Ruspini,H. (ed.), *Proceedings of the IEEE International Conference on Neural Networks (ICNN 93)*. IEEE, San Francisco, pp. 586–591.

States,D.J. and Botstein,D. (1991) Molecular sequence accuracy and the analysis of protein coding regions. *Genetics*, **88**, 5518–5522.

Xu,Y., Mural,R.J. and Uberbacher,E.C. (1995) Correcting sequencing errors in DNA coding regions using a dynamic programming approach. *Comput. Appl. Biosci.*, **11**, 117–124.

Zell,A., Mache,N., Hübner,R., Mamier,G., Vogt,M., Herrmann,K.U., Schmalzl,M., Sommer,T., Hatzigeorgiou,A., Döring,S., Posselt,D., Reczko,M. and Riedmiller,M. (1993) SNNS user manual, Version 3.0. *Technical Report*, Universität Stuttgart, Fakultät Informatik.