Indian Journal of Biochemistry & Biophysics
Vol. 48, December 2011, pp. 416-421

Automatic prediction of non-coding RNA genes in prokaryotes based on compositional statistics

Hao Tong, Feng-Biao Guo*and Yuan-Nong Ye

School of Life Science and Technology, University of Electronic Science and Technology of China,
Chengdu 610054, China

Received 30 May 2011; revised 27 October 2011

Although non-coding RNA (ncRNA) genes do not encode proteins, they play vital roles in cells by producing functionally important RNAs. In this paper, we present a novel method for predicting ncRNA genes based on compositional features extracted directly from gene sequences. Our method consists of two Support Vector Machine (SVM) models — Codon model which uses codon usage features derived from ncRNA genes and protein-coding genes and Kmer model which utilizes features of nucleotide and dinucleotide frequency extracted respectively from ncRNA genes and randomly chosen genome sequences. The 10-fold cross-validation accuracy for the two models is found to be 92% and 91%, respectively. Thus, we could make an automatic prediction of ncRNA genes in one genome without manual filtration of protein-coding genes. After applying our method in *Sulfolobus solfataricus* genome, 25 prediction results have been generated according to 25 cut-off pairs. We have also applied the approach in *E. coli* and found our results comparable to those of previous studies. In general, our method enables automatic identification of ncRNA genes in newly sequenced prokaryotic genomes. Datasets and program code used in this work are available at http://cobi.uestc.edu.cn/resource/SS_ncRNA/

Keywords: Automatic gene prediction, Non-coding RNA genes, *Sulfolobus solfataricus, E. coli,* Nucleotide composition, Support vector machine

Non-coding RNA (ncRNA) genes encode functional RNA molecules, which are not translated into proteins. However, these functional molecules actually play essential roles in cellular process ranging from regulation of gene expression[1] to RNA modification[2]. Thus, identification of (ncRNA genes) may help to elucidate gene regulatory network[3,4]. However, their identification in a particular genome is still challenging. On one hand, the capabilities of experimental methods for identifying ncRNAs are limited. On the other hand, lacking of significant signals like open-reading-frames makes it difficult to predict ncRNA genes with computational approaches developed for prediction of protein-coding genes.

However, despite these difficulties, a number of computational methods have been developed[5-13] for predicting ncRNA genes. These methods are based on either comparative genomics[5,6,10,11] or statistical theory[7,9]. Washietl et al.[5] reported a method, which combines comparative sequence analysis and structure prediction for detecting functional RNAs. Rivas et al.[6] presented a computational comparative genomic screen for ncRNA genes. Their key idea has been to distinguish conserved RNA secondary structures from a background of other conserved sequences using probabilistic models of expected mutational patterns in pair-wise sequence alignments. Among statistical theory methods, Schattner[7] developed a screening program to identify ncRNA genes using local variations in single-base and dinucleotide statistics. Tran et al.[9] presented a *de novo* prediction algorithm for ncRNA genes using statistical parameters derived from sequences and structures of known ncRNA genes in comparison to negative controls.

Although the comparative genomics approach is successful, the limitations still remain. First, it will miss the unique ncRNAs in specific genomes, which do not have homologues in other species. Furthermore, approaches of comparative genomics rely greatly on high quality alignments. An alternative approach focuses on features derived from ncRNAs themselves and, therefore, requires no homology information. Algorithms based on both compositional[14] and structural[9] features have achieved success; however, they either limit searching ncRNAs

*Author for correspondence.
Fax: +86 28 83208238
E-mail: fbguo@uestc.edu.cn

to intergenic regions[14] or rely on manual filtering of the ncRNA gene candidates which overlap the protein-coding genes for a certain number of nucleotides[9].

Here, we report a novel automatic prediction algorithm to predict ncRNA genes in prokaryotic genomes without manual filtration and alignments. Unlike other researchers who used only the intergenic sequences[14,15], shuffled ncRNAs[9,16,17] or the randomly shuffled protein-coding genes[18], we have added protein-coding sequences as a negative set in one of our two models. We then extract the compositional features to train Support Vector Machine (SVM) to generate two different classification models, which are subsequently applied to the *Sulfolobus solfataricus* genome. The final prediction results are generated by combining the results from the two individual predictions above.

Methods

Because of its time efficiency in dealing with high-dimension data and effectiveness in classification problems[19,20], we decided to choose SVM (Support Vector Machine) as the classifying method. LIBSVM[21] was downloaded to execute the SVM experiments. To train the two classification models, we i) generated the positive and negative sets, both of which consisted of gene sequences, ii) extracted each sequence's compositional features, which could distinguish the positive data sets and the negative ones, and iii) used the features to train the SVM to generate the classification models.

Data set preparation

We first downloaded the file of ncRNA genes from the following website: http://csbl.bmb.uga.edu/publications/materials/tran/[9]. The original 936 non-redundant ncRNAs (tRNA genes and rRNA genes were not included) were collected from three sources: i) the NONCODE database[22], ii) published literature, and iii) GenBank database. For the Codon model, we randomly selected 200 ncRNAs among the original 936 non-redundant ncRNAs as the positive set. And they belonged to over 50 prokaryotic organisms, which did not include *S. solfataricus*. When choosing a ncRNA gene from one specific organism, we also selected a protein-coding gene, which was 3-times as long as the ncRNAs from that organism. Then we chose the middle part that was 1/3 of the selected protein-coding gene to make up the negative set. Therefore, sequences in the positive and negative sets had the same length distribution. We removed the ends of the protein-coding gene as these two regions could be the targets of microRNAs, which regulate the gene expression by complementarities[2]. The fact that ncRNAs could partially overlap the 5’and 3’end of protein-coding gene[23] also justified choosing the middle part as the negative control.

For the second model, we chose all sequences that satisfied two criteria: (i) they were longer than 160 bp, and (ii) they were not included in *S. solfataricus* from the original 936 ncRNAs to construct the positive set. The negative set was generated according to the strand and length information of ncRNAs: for one ncRNA gene with specific length and strand information in the positive set, we randomly selected a sequence with the same length from the same strand of *S. solfataricus* genome.

We used two different data sets for two ncRNA gene prediction models because two models were used to eliminate two different types of ncRNA candidates which were in fact not ncRNA. The Codon model was used to eliminate protein-coding genes; therefore, the negative set in this model consisted of 200 protein-coding genes. The Kmer model was used to exclude common non-ncRNA sequences, thus the negative control in this model included randomly selected background genome sequences. The results of two prediction models were combined in order to discard all kinds of non-ncRNA sequences.

Features used

In order to identify ncRNA genes, the prediction models have to be able to recognize potential ncRNA genes and then filter out non-ncRNA sequences like protein-coding genes and intergenic sequences. Therefore, the features used in prediction models were critical for ncRNA identification.

For the first model or Codon model, we examined the abilities of compositional features, including mono and dinucleotide frequencies and codon usage to distinguish ncRNAs from protein-coding genes. After comparison, the codon usage was chosen as the discriminating feature, since significant differences of codon usage between ncRNAs and protein-coding genes were observed in our study. And those differences did not limit to just one genome, that is to say, the discrepancies still existed when sequences in both positive and negative sets were collected from a large number (over 50) of bacterial organisms.

Although only protein-coding genes have codon usage and not the ncRNAs, we still named the first model Codon model for convenience in this paper because the manner of calculating the features for ncRNAs was similar with that for protein-coding genes. By stating codon model, we did not mean that ncRNAs also have codon; codon usage just indicates the algorithm of feature computation.

For the second model, frequencies of mono- and di-nucleotides were utilized, since they were shown to be useful for separating ncRNAs from background sequences in previous work[9, 14]. We also used k-mers with k = 3, 4 and 5, but that did not significantly improve the prediction accuracy.

SVM model training

Once the training samples were prepared, the prediction model was created by executing the SVMLIB training program---svm-train.exe with the default parameters.

Codon model

We assigned two different labels (0 and 1) to ncRNAs and protein-coding sequences and combined their codon usage parameters into a training set. After optimizing the SVM parameters C and Gamma by running grid.py[21] program in the training step, an accuracy of 92% was obtained under 10-fold cross-validation, which further indicated that codon usage features could significantly separate the positive (ncRNAs) and negative (protein-coding gene) sets. Finally, we obtained the classification model with the optimized parameters by executing the command: "svm-train -b 1 -c xx -g xx training_set_file".

Kmer model

We assigned 0 & 1 labels to ncRNAs and randomly chosen genome sequences, and then combined their mono- and di-nucleotide frequency parameters into a training_set_file, analogous to the Codon model. We also optimized the SVM parameters C and Gamma and then run the svm-train program to get the Kmer Model, which had an accuracy of 91% for 10-fold cross validation in discriminating ncRNAs from genome sequences.

In the next stage, the above two trained models were used to predict a window sequence as ncRNA gene or not. Only those windows classified as positive by both models are retained as recognized ncRNAs.

Results

Application to whole genome prediction

In order to apply our method we chose the organism *S. solfataricus* (NC_002754) in which sliding window technique was adopted to slide through the whole Watson and Crick strands with the window length of 160 bp and the step size of 80 (160/2) bp.

Using Codon model to filter protein-coding genes

For each sliding window, we calculated the frequencies of all 64 codons. To account for the fact that a gene could be encoded in any of the three possible reading frames, we implemented the following strategy: i) calculated the codon usages of three frames for each sliding window; ii) classified those three sequences using the trained Codon SVM model, and iii) if any of the three sequences were classified as a protein-coding gene then the original sliding window was not deemed as a ncRNA gene. This strategy ensured that the correct frame was not missed when the sliding window overlapped with a protein coding-gene.

Using Kmer model to eliminate common non-ncRNA sequences

We calculated the 20 statistics, namely 4 mono- and 16 di-mer frequencies for each sliding window in the *S. solfataricus* genome. Then we assigned to each window a positive (ncRNA gene) or negative prediction based on the Kmer SVM model generated above.

Combining the prediction results generated by the two models

A sliding window sequence was deemed a candidate ncRNA gene only if it was classified as positive in both the models. To refine the prediction results further, we used the probability estimate option of LIBSVM[21]and then set 5 different cut-offs: 50% to 90% with 10% step for each model. By combining the refined prediction results of the two models at each cut-off-pair level, we obtained 25(5*5) new prediction results. The final predictions were obtained by joining adjacent windows that were classified as positive in the new predictions. The finally recognized ncRNAs were considered correct indeed, if they overlapped with known non-coding RNA genes in the same strand.

Statistical evaluations

For each pair of cut-offs, we obtained the recognized ncRNA genes, which were referred as ‘Predicted’. Then we calculated the corresponding sensitivity (Sn), which measures the algorithm's capacity to predict the ‘True Positives’ and ‘Positive Prediction Value’ (PPV) for measuring the method's ability to exclude ‘False Positives’ based on the known non-coding RNA genes in *S. solfataricus*, as

Table 1—Whole genome prediction results of *S. solfataricus* at different level of cut-off*

KMER_M._ CUTOFF	CODON_M._ CUTOFF	PREDICTED_ NO	Sn (%)	PPV
50	50	7672	84.4	0.008
	60	7176	81.8	0.009
	70	6403	71.4	0.009
	80	5150	67.5	0.010
	90	3230	54.5	0.013
60	50	7503	79.2	0.008
	60	6930	76.6	0.009
	70	6103	66.2	0.008
	80	4851	61.0	0.009
	90	3019	49.4	0.013
70	50	7175	74.0	0.008
	60	6545	72.7	0.009
	70	5707	66.2	0.009
	80	4500	61.0	0.010
	90	2781	49.4	0.014
80	50	6757	71.4	0.008
	60	6083	70.1	0.009
	70	5233	64.9	0.009
	80	4077	61.0	0.010
	90	2511	49.4	0.015
90	50	5828	64.9	0.009
	60	5153	64.9	0.009
	70	4365	59.7	0.011
	80	3363	53.2	0.012
	90	2052	42.9	0.016

*KMER_M._CUTOFF and CODON_M._CUTOFF refers the probability cut-off of Kmer and Codon models, respectively, while PREDICTED_NO., Sn and PPV stand for the number of recognized ncRNAs, sensitivity and positive prediction value.

Table 2—Statistics of correctly found ncRNAs with higher GC/AT at different cut-off pair in *S. solfataricus**

KMER_M._ CUTOFF	CODON_M._ CUTOFF	Higher_GC_ RATE (%)	Higher_AT_ RATE (%)
50	50	90.7	76.5
	60	88.4	73.5
	70	74.4	67.6
	80	72.1	61.8
	90	60.5	47.1
60	50	86.0	70.6
	60	83.7	67.6
	70	69.8	61.8
	80	65.1	55.9
	90	53.5	44.1
70	50	81.4	64.7
	60	79.1	64.7
	70	69.8	61.8
	80	65.1	55.9
	90	53.5	44.1
80	50	79.1	61.8
	60	76.7	61.8
	70	69.8	58.8
	80	65.1	52.9
	90	53.5	41.2
90	50	74.4	52.9
	60	74.4	52.9
	70	67.4	50.0
	80	60.5	44.1
	90	51.2	32.4

*KMER_M._CUTOFF and CODON_M._CUTOFF have the same meanings as in Table 1. Higher_GC/AT_ RATE is the ratio of the number of correctly found ncRNAs with higher GC/AT content to the total number of known ncRNAs with higher GC/AT content in *S. Solfataricus.*

shown in Table 1. In this table, KMER_M._CUTOFF and CODON_M._CUTOFF refers the probability cut-off of Kmer and Codon models, respectively, while PREDICTED_NO., Sn and PPV stand for the number of recognized ncRNAs, sensitivity and positive prediction value.

Results analysis

In Table 1, with the cut-off rising there was a general trend for Sn to decrease, while the PPV tended to increase. Therefore, users can select a threshold pair to get a desired Sn and PPV trade-off for their application.

In order to investigate how the G + C contents of ncRNAs relate to the prediction results, we listed the percentages of correctly found ncRNAs with higher GC or AT content (GC or AT content >50%) for each cutoff pair (Table 2). In Table 2, KMER_M._CUTOFF and CODON_M._CUTOFF also stands for the probability cut-off of Kmer and Codon models, respectively. Higher_GC/AT_ RATE is the ratio of the number of correctly found ncRNAs with higher GC/AT content to the total number of known ncRNAs with higher GC/AT content in *S. Solfataricus*. The total number of known ncRNAs with higher GC content (> 50%) in *S. solfataricus* is 43 and the total number of known ncRNAs with higher AT content (> 50%) in *S. solfataricus* is 34. It can be noted that each Higher_GC_ RATE exceeded the Higher_AT_ RATE from 8 to 21.5%. Thus, we can conclude that ncRNAs with higher GC contents were relatively easier to be correctly found than those with higher AT contents in *S. solfataricus*. This was possibly due to the low overall GC content of *S. solfataricus* genome (35%).

We also applied the algorithm without changing training set and trained models to *E. coli*, an organism that has been analyzed by many methods for prediction of ncRNA genes. As PPV did not change dramatically among different cut-off pairs, we decided to choose the prediction result at the cut-off

Table 3—Prediction accuracies by different programs for *E. coli**

Program	No. of predictions	Sn (%)	PPV
Rivas	275	40.9	0.138
Wang	420	7.5	0.017
Tran	601	40.9	0.063
Tong	4816	51.1	0.010

*Number of predictions, sensitivity [Sn = TP/(TP + FN)] and positive prediction value [PPV = TP/(TP + FP)] are given for each program[6,9,18], where TP and FN denote the numbers of real ncRNAs that have been predicted as ncRNAs and non-ncRNAs. FP denotes the number of sequences that have been falsely predicted as ncRNAs.

Table 4—Prediction results of only one SVM model in *S. solfataricus**

Cut-off	No. of predictions	Sn (%)	PPV
50	4431	54.5	0.009
60	2973	41.6	0.011
70	1880	33.8	0.014
80	1066	24.7	0.018
90	437	15.6	0.027

*Cut-off denotes the cut-off values in the single SVM model. The number of predictions, sensitivity [Sn = TP/(TP + FN)] and positive prediction value [PPV = TP/(TP+ FP)] is given for each program, where TP and FN denote the numbers of real ncRNAs that have been predicted as ncRNAs and non-ncRNAs. FP denotes the number of sequences that have been falsely predicted as ncRNAs.

level of 50 and 50 to make a comparison with former methods for *E. coli*. Compared with the two earlier ncRNA predictions[6,9] in *E. coli*, as can be seen from Table 3, our method had significantly higher sensitivity (51.1% *versus* 40.9% and 40.9%); while the PPV was relatively low (0.010 to 0.138 and 0.017). This might be due to the fact that their program relied on prior knowledge of multiple alignments for identification of conserved regions[6] or because they filtered protein coding regions manually[9]. When compared to Wang's work[18], our approach had a significant improvement in sensitivity (7.5% and 51.1%, respectively) with approximately the same PPV (0.010 and 0.017). Another point worth noting in the prediction result was that we correctly found an ncRNA gene (4188065, 4187905, '-') that was fully embedded within a protein-coding gene (4187809, 4188348, '+'), which might be due to the automatic filtering step adopted in our method.

Discussion

In this study, we developed an algorithm based on SVM, which utilized compositional features for identifying ncRNA genes in prokaryotes. We successfully applied it to the organisms *S. solfataricus* and *E. coli*. Our method did not require homology or conservation information and the high quality alignments. Using the various kinds of ncRNAs from diverse organisms as training set enabled us to correctly predict ncRNAs in a newly sequenced prokaryotic genome without prior information of genes (both ncRNAs and protein-coding genes) in that genome.

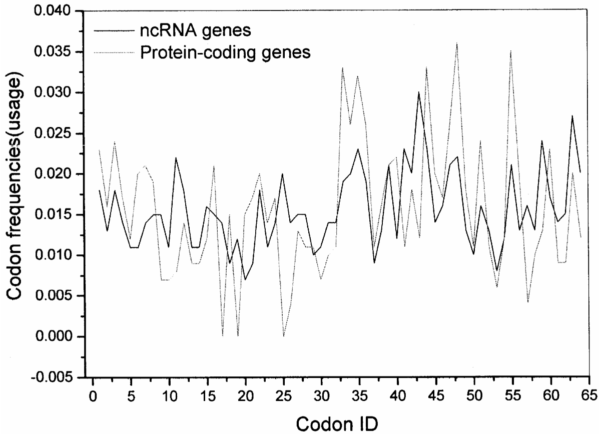

Fig. 1—Average frequencies of all 64 codons for 200 ncRNA genes and 200 protein-coding genes in the training set of the Codon model [As can be seen, the codon usage of protein-coding genes tends to be more undulated than that of ncRNAs, even the sources of ncRNAs and protein-coding genes are over 50 prokaryotic organisms. Therefore, this type of codon usage differences could be used as the theoretical foundation of Codon model for discarding protein-coding genes]

An interesting finding in our work is shown in Fig. 1. For this figure, ncRNA genes and protein-coding genes were all from the training set of the Codon model, i.e. the 200 ncRNAs and 200 protein-coding genes and the frequency of each codon was the average of 200 sequences both for ncRNA genes and protein-coding genes. It can be noticed that the codon usage of protein-coding genes tended to be more undulated than that of ncRNAs in various prokaryotic organisms. It was possibly because of the codon bias[24] in protein-coding genes, which is influenced by complex factors. Absence of such bias makes the codon usage of ncRNAs distribute more randomly. Therefore, this type of codon usage differences was utilized in the Codon model in our approach.

Besides the method factually adopted here, there is the other way to construct classification models, i.e. combining all features-mono-, dimer frequencies and codon usage in a single SVM model and also combining the two training sets into one trains set. Thus, only one single model is needed to train in this alternative method. We tested this approach and the result is shown in Table 4. We set the cut-off from 0.5 to 0.9 and counted the predicted ncRNAs, the Sn and

PPV for each cut-off. Apparently, this result was not as good as that generated by the two separate models. When they had the same PPV, the Sn of the separate model was approximately 20% higher than that of the single model. Although the single model had the highest PPV of 0.027 at the 0.9 cut-off, the Sn (15.6%) was too low to be acceptable.

The model combination strategy can be regarded as a method with two continuous procedures; thus we can assume that the whole genome goes through the Kmer and Codon models in succession when the two separate prediction results are combined. Kmer model is used to discard background sequences; while adding Codon model can further filter out the protein-coding sequences in the prediction results of the Kmer model. Thus, the intersection of prediction results of two models can make the gene finding process automatic without manual filtration of protein-coding sequences[9].

The features used in algorithm: mono-and dimer frequencies and codon usage could be easily obtained from biological sequences, suggesting that our method could be implemented and used easily. However, it should be noted that secondary structure-based features, e.g., Z-score, structure conservation index[25], pattern of substitution[26], folding free energy[14] and ensemble statistics[9] are very effective in ncRNA gene identification. We also investigated some of those features, however, improvements were too tiny to mention when applying to *S. solfataricus* (data not shown). This was possibly because that its low genome GC content (35%) made the compositional features alone be efficient to correctly identify ncRNAs in this genome. To apply the algorithm in a consistent form, we neglected structure-based features in application of *E. coli*.

Acknowledgements

We thank Dr. Fengfeng Zhou at University of Georgia and Professor Hao Lin at UESTC for useful discussions. We also thank Mr. Lu-Wen Ning for helps in providing the web service for this work. The present study was supported by National Natural Science Foundation of China (grant 31071109 and 60801058), and the special fund of China Post-doctoral Science Foundation (Grant 201104687) and the program for New Century Excellent Talents in University (NCET-11).

References

1 Saetrom P, Snove O, Nedland M, Grunfeld T B, Lin Y, Bass M B & Canon J R (2006) *Oligonucleotides* 16, 115-144

2 Eddy S R (2001) *Nat Rev Genet* 2, 919–929

3 Mattick J & Makunin I (2006) *Hum Mol Genet* 15, R17–R29

4 Ke X S, Liu C M, Liu D P & Liang C C (2003) *Curr Opin Chem Biol* 7, 516–523

5 Washietl S, Hofacker I L & Stadler P F (2005) *Proc Natl Acad Sci* (USA) 102, 2454–2459

6 Rivas E, Klein R J, Jones T A & Eddy S R (2001) *Curr Biol* 11, 1369-1373

7 Schattner P (2002) *Nucleic Acids Res.* 30, 2076–2082

8 Schattner P, Barberan-Soler S & Lowe T M (2006) *RNA* 12, 15-25

9 Tran T T, Zhou F F, Marshburn S, Stead M, Kushner S R & Xu Y (2009) *Bioinformatics* 25, 2897-2905

10 Wassarman K M, Repolia F, Rosenow C, Storz G & Gottesman G (2001) *Genes Dev* 15, 1637–1651

11 Argaman L, Hershberg R, Vogel J, Bejerano G, Wagner E G, Margalit H & Altuvia S (2001) *Curr Biol* 11, 941–950

12 Klein R J, Misulovin Z & Eddy S R (2002) *Proc Natl Acad Sci* (USA) 99, 7542–7547

13 Zhang Y, Zhang Z, Ling L, Shi B & Chen R (2004) *Bioinformatics* 20, 599–603

14 Carter R J, Dubchak I & Holbrook S R (2001) *Nucleic Acids Res* 29, 3928-3938

15 Saetrom P, Sneve R, Kristiansen K I, Snove O, Jr Grunfeld T, Rognes T & Seeberg E (2005) *Nucleic Acids Res* 33, 3263–3270

16 Clote P, Ferré F, Kranakis E & Krizanc D (2005) *RNA* 11, 578–591

17 Rivas E & Eddy S R (2000) *Bioinformatics* 16, 583–605

18 Wang C, Ding C, Meraz R F & Holbrook S R (2006) *Bioinformatics* 22, 2590–2596

19 Chang C C, Hsu C W & Lin C J (2000) *IEEE Trans NEURAL NETWOR* 11, 1003-1008

20 Liu J, Gough J & Rost B (2006) *PLoS Genet* 2, e29 doi: 101371/journal pgen002002

21 Chang C C & Lin C J (2001) *ACM Trans. Intell Sys Techol*, 2, 27:1--27:27, Software available at http://www.csie.ntu.edu.tw/~cjlin/libsvm

22 Liu C, Bai B, Skogerbo G, Cai L, Deng W, Zhang Y, Bu D, Zhao Y & Chen R (2005) *Nucleic Acids Res* 33, D112–D115

23 Gaspin C, Cavaille J, Erauso G & Bachellerie J P (2000) *J Mol Biol* 297, 895–906

24 Ermolaeva M D (2001) *Curr Issues Mol Biol* 3, 91-97

25 Gruber A R, Findeiß S, Washietl S, Hofacker I L & Stadler P F (2010) *Pacific Symp Biocomputing* 15, 69-79

26 Rivas E & Eddy S R (2001) *BMC Bioinformatics* 2, 8