Identification of Promoter Region in Genomic DNA Using Cellular Automata Based Text Clustering

Kiran Sree[1] and Ramesh Babu[2]
[1]Department of Computer Science, Jawaharlal Nehru Technological University, India
[2]Department of Computer Science, Acharya Nagarjuna University, India

Abstract: *Identifying the promoter regions play a vital role in understanding human genes. This paper presents a new cellular automata based text clustering algorithm for identifying these promoter regions in genomic DNA. Experimental results confirm the applicability of cellular automata based text clustering algorithm for identifying these regions. We also note an increase in accuracy of fining these promoter regions by 12 percent for DNA sequences for shorter length. This algorithm was trained to identify promoter regions in mixed and overlapping DNA sequences also. However this algorithm fails in identifying the promoter regions of length greater than 54. This algorithm will be also used to predict the RNA structure.*

Keywords: *Text clustering, DNA sequence, cellular automata.*

Received May 10, 2008; accepted October 1, 2008

1. Introduction

Many of the challenges in biology are now challenges in computing. Bioinformatics [16], the application of computational techniques to analyze the information associated with bimolecules on a large scale, has now firmly established itself as a discipline in molecular biology. Bioinformatics is a management information system for molecular biology. Bioinformatics encompasses everything from data storage and retrieval to the identification and presentation of features within data, such as finding genes within DNA sequence, finding similarities between sequences, structural predictions. Analyzing the coding regions is not the scope of this research.

Finding the promoter regions is difficult because, genomes are small are not continuous (0.1-10.1bp).The coding density [1] in eukaryotes is less than 50% till date (only 5% of genes constitutes of coding regions). From the Figure 1 we now that DNA is arranged in the form of exons and introns. Introns form maximum part of DNA and extons form the minor part of DNA.

But, Proteins attach to the DNA and help the DNA strands coil up into a chromosome when the cell gets ready to divide. The point where the proteins add to the strands is very important to this research. We also need to find the closet neighbour hood stride, which is the main reason for the cause of dynamism in this research project. We use trust region method for finding the optimized stride [15]. Once we find the neighbouring strides we use parallel scan algorithm for processing the sequence to find the coding region.

2. Cellular Automata

Cellular automata use localized structures to solve problems in an evolutionary way. Cellular Automata (CA) often demonstrates also significant ability toward self-organization that comes mostly from the localized structure on which they operate. By organization, one means that after some time in the evolutionary process, the system exhibits more or less stable localized structures. This behaviour can be found no matter the initial conditions of the automata.

A CA consists of a number of cells organized in the form of a lattice. It evolves in discrete space and time. The next state of a cell depends on its own state and the states of its neighbouring cells. In a 3-neighborhood dependency, the next state $q_i(t + 1)$ of a cell is assumed to be dependent only on itself and on its two neighbours (left and right) and is denoted as:

$$q_i(t + 1) = f\,(q_{i-1}(t),\ q_i(t),\ q_{i+1}(t)) \qquad (1)$$

where, $q_i\ (t)$ represents the state of the ith cell at tth instant of time, f is the next state function and referred to as the rule of the automata. The decimal equivalent of the next state function, as introduced by Wolfram, is the rule number of the CA cell·

2.1. Fuzzy CA Fundamentals

Fuzzy Cellular Automata (FCA) is a linear array of cells which evolves in time. Each cell of the array assumes a state *qi*, a rational value in the interval [0, 1] (fuzzy states) and changes its state according to a local evolution function on its own state and the states of its two neighbours. The degree to which a cell is in fuzzy states 1 and 0 can be calculated with the membership functions. This gives more accuracy in identifying the promoter region. In a FCA, the conventional Boolean functions are AND, OR, NOT.

2.2. Dependency Matrix for FCA

Rules defined in equation 1 should be represented as a local transition function of FCA cell. That rules as shown in Table 1 are converted into matrix form for easier representation of chromosomes.

$$T = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 \end{bmatrix}$$

Figure 1. Matrix representation of rule.

Example 1: a 4-cell null boundary hybrid FCA with the following rule <238, 254, 238, 252> (that is, <(q_i+q_{i+1}), (q_{i-1}+q_i+q_{i+1}), (q_i+q_{i+1}), (q_{i-1}+q_i)>) applied from left to right, may be characterized by the following dependency matrix.

Table 1. Rule of FCA.

Non- complemented rules		Complemented rules	
Rule	**Next state**	**Rule**	**Next state**
0	0	255	1
170	q_{i+1}	85	$\overline{q}i + 1$
204	q_i	51	$\overline{q}i$
238	q_i+q_{i+1}	17	$\overline{qi + qi + 1}$
240	q_{i-1}	15	$\overline{q}i - 1$
250	q_{i-1}+q_{i+1}	5	$\overline{qi - 1 + qi + 1}$
252	q_{i-1}+q_i	3	$\overline{qi - 1 + qi}$

While moving from one state to other, the dependency matrix indicates on which neighbouring cells the state should depend. So cell 254 depends on its state, left neighbour and right neighbour.

Now we represented the transition function in the form of matrix as shown in Figure 1. In the case of complement, FCA we use another vector for representation of chromosome. Rules will be used for inducing dynamism into our project.

3. Genetic Algorithm and Cellular Automata

The main motivation behind the evolving cellular automata framework is to understand how Genetic Algorithms (GA) evolve cellular automata that perform computational tasks requiring global information processing, since the individual cells in a CA can communicate only locally without the existence of a central control the GA has to evolve CA that exhibit higher-level emergent behaviour in order to perform this global information processing.

This framework provides an approach to studying how evolution can create dynamical systems in which the interactions of simple components with local information storage and communication give rise to coordinated global information processing.

4. Text Clustering with Cellular Automata

The algorithm is composed of the following steps:

Place K points into the space represented by the basins that are being clustered. These points represent initial group of CA Evaluation Parameters.

Assign each basin to the group that has the closest Basin.

When all basins have been assigned, recalculate the positions of the K CA evaluation parameters.

Repeat steps 2 and 3 until the CA evaluation parameters no longer move. This produces a separation of the basins into groups from which the metric to be minimized can be calculated.

Although it can be proved that the procedure will always terminate, the *k*-means algorithm does not necessarily find the most optimal configuration, corresponding to the global function minimum. The algorithm is also significantly sensitive to the initial randomly selected cluster centres. The *k*-means algorithm can be run multiple times to reduce this effect.

Input: text clustering with CA (constraints).
Output: CA based inverted basins.

```
Start generate a CA with k distinct number of CA basins
distribute the parameters into k CA basins.
evaluate the distribution in each closet basin.
calculate the Rm (closet basin).
swap the more appropriate features to the bottom
leaves of the inverted basin tree.
stop.
```

5. Experimental Results

The below tables show the predictive accuracy of different algorithms on identifying both promoter and non-promoter regions in DNA sequences. In this section we present the results of CATC classifier for several datasets. Values are given for the percentage accuracy on test set promoter sequences and the percentage accuracy on test set non promoter sequences

CATC is used to find promoter regions for all sequence lengths. But the accuracy will be more for DNA Sequences of lesser length. The percentage accuracy reported was 62.3%. Figure 2 shows that CATC can be used to identify protein promoter regions and the results obtained were comparable with other standard algorithms. CATC overcome all the disadvantages of previous standard algorithms like fixing the position of the gene and static order of the DNA sequence.

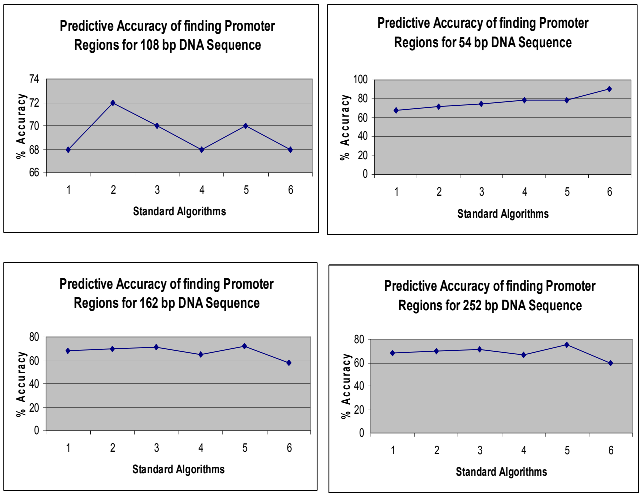


Figure 2. Predictive accuracy.

Table 2. Predictive accuracy for length 54 human DNA sequence.

Algorithm	Promoter	Non Promoter
Dicodon Usage	61%	57%
Bayesian	51%	46%
SUCA	78%	72%
UN FMACA	79%	72%
NPCRIT	79.5%	72.8%
CATC	80.2%	71.6%

Table 3. Predictive accuracy for length 252 human DNA sequence.

Algorithm	Promoter	Non Promoter
Dicodon Usage	68%	64%
Bayesian	52%	46%
SUCA	58%	62%
UN FMACA	59%	50%
NPCRIT	69.5%	62.8%
CATC	48.22%	46%

Table 4. Predictive accuracy for length 102 human DNA sequence.

Algorithm	Promoter	Non Promoter
Dicodon Usage	58%	54%
Bayesian	50%	46%
SUCA	68%	71%
UN FMACA	69%	50%
NPCRIT	79.5%	62.8%
CATC	68.22%	65.6%

Table 5. Predictive accuracy for length 162 human DNA sequence.

Algorithm	Promoter	Non Promoter
Dicodon Usage	58%	54%
Bayesian	50%	46%
SUCA	58%	60%
UN FMACA	59%	50%
NPCRIT	69.5%	62.8%
CATC	58.22%	56%

6. Conclusion

In this paper, we present a new Cellular automata based text clustering algorithm for identifying the promoter regions in genomic DNA. It was also applied to predict the structure of RNA. On small DNA sequences, our algorithm was found very effective. This algorithm was not yielding good results with DNA sequence length more than 54. This proposed algorithm can also be also extended to find protein coding regions in genomic DNA also.

Referances

[1] Chattopadhyay S., Adhikari S., Sengupta S., and Pal M., “Highly Regular, Modular, and Cascadable Design of Cellular Automata Based Pattern Classifier,” *Computer Journal of IEEE Transactions on Very Large Scale Integration* System, vol. 8, no. 6, pp. 25-29, 2000.

[2] Eric S. and Gary D., *Identification of Protein Coding Regions in Genomic DNA*, ICCS Transactions, 2002.

[3] Farber R., Lapedes A., and Sirotkin K., “Determination of Eukaryotic Protein Coding Regions Using Neural Networks and Information Theory,” *Computer Journal of Molecular Biolog*y, vol. 226, no. 3, pp. 471-479, 1992.

[4] Fickett J., “Recognition of Protein Coding Regions in DNA Sequences,” *Computer Journal of Nucleic Acids Research*, vol. 10, no. 1, pp. 5303-5318, 1982.

[5] Flocchini P., Geurts F., Mingarelli A., and Santoro N., “Convergence and Aperiodicity in Fuzzy Cellular Automata: Revisiting Rule 90,” *Computer Journal of Physica D*, vol. 4, no. 2, pp. 150-158, 2000.

[6] Gil E., Murray W., and Wright H., *Practical Optimization*, Academic Press, 1997.

[7] Kiran P., “Improving Quality of Clustering Using Cellular Automata for Information Retrieval,” *Computer Journal of Computer Science*, vol. 4, no. 2, pp. 167-171, 2008.

[8] Kiran S. and Ramachandran R., “Identification of Protein Coding Regions in Genomic DNA Using Supervised Fuzzy Cellular Automata,”

Computer Journal of Advances in Computer Science and Engineering, vol. 1, no. 2, pp. 51-60, 2008.

[9] Kiran S. and Ramesh B., "Identification of Protein Coding Regions in Genomic DNA Using Unsupervised FMACA Based Pattern Classifier," *Computer Journal of Computer Science and Network Security*, vol. 8, no. 1, pp. 305-308, 2008.

[10] Langton C., "Self Reproduction in Cellular Automata," *Computer Journal of Physica D*, vol. 10, no. 3, pp. 135-144, 2000.

[11] Larsen B. and Aone C., "Fast and Effective Text Mining Using Linear Time Document Clustering," *in Proceedings of the 5th ACM SIGKDD Interna1ional Conference on Knowledge Discovery and Data Mining*, San Diego, pp. l6-22, 1999.

[12] MaCQueen J., "Some Methods for Classification and Analysis of Multi Attribute Instances 15th Berkeley Symposium on Mathematics," *Computer Journal of Statistics and Probability*, vol. 5, no. 4, pp. 25-29, 1967.

[13] Maji P. and Chaudhuri P., "FMACA: A Fuzzy Cellular Automata Based Pattern Classifier," *in Proceedings of 9th International Conference on Database Systems*, Korea, pp. 494-505, 2004.

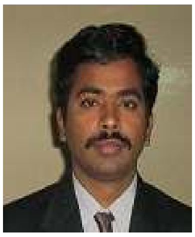
[14] Maji P. and Chaudhuri P., "Fuzzy Cellular Automata for Modeling Pattern Classifier," *Computer Journal of Accepted for Publication in IEICE*, vol. 25, no. 6, pp. 142-146, 2004.

[15] Maji P., "FMACA: A Fuzzy Cellular Automata Based Classifier," *in Proceeding of 9th International Conference on Database Systems*, Korea, pp. 494-505, 2002.

[16] Stéphane A. and Jean C., "Self Identification of Protein Coding Regions in Microbial Genomes," *Centre National de la Recherche Scientifique EP.91-95*, 2002.

[17] Toffoli T., *Reversible Computing in Automata Languages and Programming*, Springer, New York, 1980.

[18] Vichniac G., "Simulating physics with Cellular Automata," *Computer Journal of Physica D*, vol. 10, no. 5, pp. 96-115, 1994.



Kiran Sree received his BTech in computer science and engineering, from J.N.T.U and ME in computer science and engineering from Anna University. He is pursuing PhD in computer science from J.N.T.U, Hyderabad. He was the Associate Editor for six international journals.

Ramesh Babu received his BE in electronics and communication engineering from University of Mysore, ME in computer engineering from Andhra University, PhD in computer science from Acharya Nagarjuna University, He is currently working as Professor in the Department of Computer Science, Nagarjuna University.