ISSN (Online) 2394-2320

International Journal of Engineering Research in Computer Science and Engineering (IJERCSE)

Vol 10, Issue 5, May 2023

A Probabilistic Approach to Mapping DNA Overlaps to Uncover Hidden Gene Expressions

[1] Shrinithi Natarajan, [2] Dr. Jhinuk Chatterjee

[1] Computer Science and Engineering, PES University, Bangalore, India
[2] Department of Biotechnology, PES University, Bangalore, India
Corresponding Author Email: [1] kavinattu2001@gmail.com, [2] jhinukchatterjee@pes.edu

***Abstract*— Genes make proteins according to a simple genetic code but not all proteins are made in every single cell. It appears we have another layer of coding to tell a cell what proteins to make and when to make those proteins. The nucleosomes which form because of compact wrapping of DNA around histones does this extra coding. A function of the nucleosomes apart from compacting DNA is hiding genes from the cell. The spools can find the right places on the DNA to hide all the unnecessary genes. We have aimed to find a probabilistic approach to map or trace the pattern that the sequencing of the genome most closely follows so that we can use this very path to mask or unmask a gene expression. We used the Viterbi algorithm to trace the same.**

***Keywords*—genes, proteins, Viterbi Algorithm, hidden genes, probabilistic.**

I. INTRODUCTION

When the first DNA genome was sequenced by Frederick Sanger in 1977, the results solved a mystery. Previous analysis of the proteins produced by the bacteriophage (φ x174) during infection seemed to require coding sequences (CDS) longer than the measured length of the phage genome! This mystery was solved by analysis which revealed extensive overlap between coding regions with the internal scaffolding gene overlapping the genome replication gene. Such a gene overlap in eukaryotes occurs when at least one nucleotide is shared between the outermost boundaries of the primary transcripts of two or more genes, such that a DNA base mutation at the point of overlap would affect transcripts of all the genes involved in that overlap. [Wright et al, 2022] Gene overlap in prokaryotes occurs when the CDSs of two genes share a nucleotide either on the same or opposite strands.

While we can identify the location of the gene overlaps in each genome sequence, it is not enough information for us to determine the path taken for the gene expression to occur. This path can be traced effectively using the Viterbi Algorithm. We demonstrate this algorithm on *E. cuniculi*, a mammalian microbial pathogen. We identify the overlaps between the genome sequences of chromosome IV and chromosome V and using these overlaps, we trace the path followed for gene expression.

II. PREVIOUS WORKS

Here we briefly review the work done by other authors and their respective models. Bar-Joseph et al used statistical models for clustering the expressions. They also indicated that a cluster's information was also inspired by its previous states. [Bar-Joseph et al, 2002].

Friedman et al construct a Bayesian network for describing the interaction among genes. Their model takes a more temporal approach for describing regulatory interactions. [Friedman et al, 2000].

Filkov et al took yet another innovative approach to compare a pair of genes in terms of strong correlation between their expression profiles. Each time course is modelled as a piecewise linear function. The results for each pair are compared and a similarity score for each pair of the smoothened (adjusted curves) is deduced. [Filkov et al, 2002].

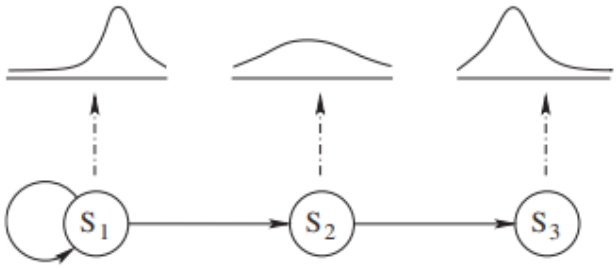


Fig. 1. A Hidden Markov Model visualized as directed graph, the emission pdfs are attached to the nodes. The model depicted is a prototype for down-regulation.

Figure 1. [Schliep et al, 2003]

As can be seen, several authors have described and demonstrated various methods for analysis of hidden gene expressions. What we haven't seen till now is that any biological sequence can simply be interpreted as a string of just 4 characters which can be repeated several times. Such a string can be modelled in any statistical model inclusive of the Viterbi algorithm as demonstrated in the following sections of the paper. It also provides the flexibility to tweak the probabilities to observe what might happen in such

ISSN (Online) 2394-2320

International Journal of Engineering Research in Computer Science and Engineering (IJERCSE)

Vol 10, Issue 5, May 2023

situations. We have converted an NLP-based HMM algorithm to something that can be applied in the field of bioinformatics.

III. MATERIALS AND METHODS

We used the genome sequences of E. cuniculi with accession numbers NC_003231.1 and NC_003232.1 for chromosome IV and chromosome V respectively. The genomic details of E. cuniculi are as follows: [Katinka et al, 2001]

- Consists of 11 linear chromosomes ranging from 217 to 315 kb
- The genome is reduced (~2.9Mb)

Table 1 General features of the *E. cuniculi* genome

Total sequenced length	2,507,519 bp
G+C content	
Protein-coding regions	47.6%
Intergenic regions	45.0%
Telomeric and subtelomeric regions	52.9%
No. of protein-coding sequences	1,997
Mean intergenic distance	129 bp
Gene density of chromosome cores	1 CDS per 1,025 bp
No. and sizes of spliceosomal introns	13 (23–52 bp)
No. of 16S–23S rRNA genes	22*
No. of 5S rRNA genes	3 (on chrV, VII, IX)
No. of tRNA genes	44†
No. and sizes of tRNA introns	2 (16, 42 bp)

*Two per chromosome.
†On all chromosomes.

Figure 2. General features of E. cuniculi genome

We used the BLAST tool to check the aligned and misaligned sequences in the two genome sequences and gaps if any.

The aligned sequence is represented in the form of a Hidden Markov Model (HMM) where the states (that are unobservable) are the chromosome sequences.

The results from the alignment data were retrieved and tested over the Viterbi Algorithm. The Viterbi algorithm is a dynamic programming algorithm for obtaining the maximum a posteriori probability estimates of the most likely sequence of hidden states called the Viterbi path that results in a sequence of observed events.

The reason for choosing the Viterbi algorithm is its dynamic nature. Moreover, compared to the brute force algorithm of searching every possible combination of nucleotides, the Viterbi Algorithm outstands the brute force algorithm (time complexity: $O(P^L)$) in that its time complexity is just $O(L*P^2)$ where L is the length of the nucleotide sequence and P is the number of different nucleotide bases that can be possible as the next base in the sequence.

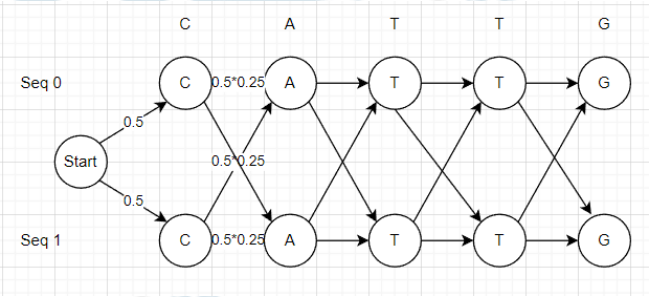
The algorithm uses the following formula for calculating the probabilities of the hidden states:

$$p(x_1 \dots x_n, y_1 \dots y_{n+1}) = \prod_{i=1}^{n+1} q(y_i|y_{i-2}, y_{i-1}) \prod_{i=1}^{n} e(x_i|y_i)$$

The brute force approach computes the probability for every possible combination of paths and picks that path which has the maximum probability. Which means that out of P different possibilities, for a sequence of length L, there will be P^L different combinations generated which can have a huge impact on the time complexity.

The Viterbi algorithm on the other hand adopts the process of eliminating that path which does not have a high probability without waiting for the final probability to be computed. The algorithm states that: given n paths that lead to the same node, that path with the maximum probability is only expanded further and the other paths are discontinued at the same time. So given P different options for a sequence of length L, there are a total of P^2 different options for one nucleotide among the L nucleotides. Hence for L nucleotides, the total number of combinations is just LP^2.

The following flow diagram converts an NLP-based algorithm in the form that is required for our study:



IV. RESULTS

A. Comparing the two chromosomes of E. cuniculi to identify the aligned sequences in the chromosomes:

BLAST was used to check for aligned and misaligned regions in the two chromosomes. To obtain a concise list of alignments along with their quality values, gaps and percent identities, Genome Workbench was used to obtain the following list:

NC_003232.1	NC_003231.1	Length	Mismatch	Gap	% Identity	Type
01-Oct	6,474-6,483	10	0	0	100	Aligned
11-Nov		1	0	1	0	Gap
12-505	6,484-6,977	494	2	0	99.6	Mixed
	6,978-6,978	1	0	1	0	Gap
506-3,631	6,979-10,104	3126	2	0	99.94	Mixed
	10,105-10,124	20	0	20	0	Gap
3,632-3,722	10,125-10,215	91	2	0	97.8	Mixed
	10,216-10,216	1	0	1	0	Gap
3,723-3,839	10,217-10,333	117	6	0	94.87	Mixed
	10,334-10,336	3	0	3	0	Gap
3,840-4,059	10,337-10,556	220	0	0	100	Aligned
	10,557-10,557	1	0	1	0	Gap
4,060-4,066	10,558-10,564	7	0	0	100	Aligned
	10,565-211,105	200541	0	200541	0	Gap
4,067-207,814		203748	0	203748	0	Gap
207,815-207,892	211,106-211,183	78	27	0	65.38	Mixed
	211,184-211,186	3	0	3	0	Gap
207,893-208,012	211,187-211,306	120	30	0	75	Mixed
208,013-208,013		1	0	1	0	Gap
208,014-208,022	211,307-211,315	9	5	0	44.44	Mixed
208,023-208,023		1	0	1	0	Gap
208,024-208,027	211,316-211,319	4	0	0	100	Aligned
	211,320-211,321	2	0	2	0	Gap

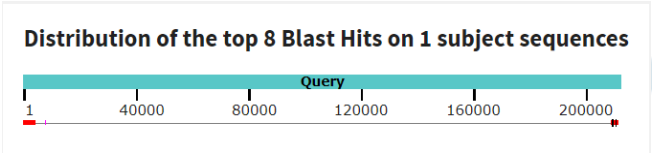
ISSN (Online) 2394-2320

International Journal of Engineering Research in Computer Science and Engineering (IJERCSE)

Vol 10, Issue 5, May 2023

208,028-208,037	211,322-211,331	10	4	0	60	Mixed
	211,332-211,334	3	0	3	0	Gap
208,038-208,096	211,335-211,393	59	25	0	57.63	Mixed
	211,394-211,396	3	0	3	0	Gap
208,097-208,132	211,397-211,432	36	13	0	63.89	Mixed
	211,433-211,433	1	0	1	0	Gap
208,133-208,142	211,434-211,443	10	4	0	60	Mixed
208,143-208,143		1	0	1	0	Gap
208,144-208,169	211,444-211,469	26	10	0	61.54	Mixed
	211,470-211,472	3	0	3	0	Gap
208,170-208,178	211,473-211,481	9	2	0	77.78	Mixed
208,179-208,181		3	0	3	0	Gap
208,182-208,234	211,482-211,534	53	11	0	79.25	Mixed
	211,535-211,536	2	0	2	0	Gap
208,235-208,237	211,537-211,539	3	1	0	66.67	Mixed
208,238-208,241		4	0	4	0	Gap
208,242-208,248	211,540-211,546	7	1	0	85.71	Mixed
	211,547-211,548	2	0	2	0	Gap

This shows the alignment regions of the nucleotide sequences and a few other minor alignment regions towards the end of the sequence. The below **Figure** ure shows the Graphic Summary of all the alignments obtained from the BlastN result.



B. Results from the Viterbi Algorithm:

The transition and emission matrix were constructed assuming that there is an equal probability of transitioning from chromosome 4 sequence to chromosome 5 sequence as well as transitioning from any of the nucleotide bases to any other base.

Assuming the above probabilities, all nucleotides seem to be a part of chromosome 4 as shown:

```
x =
 [0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0]
```

If we change the transition probabilities to see the changes, we see the following at a ratio of 1/3:2/3 transition probability:

```
x =
 [0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1]
```

The results are interpreted as follows:

‘0’ is the notation for chromosome IV sequence and ‘1’ stands for chromosome V sequence. A change from 0 to 1 or vice versa indicates a jump in the sequence that is being read.

V. DISCUSSION

The Viterbi path hence generated is just an example of the massive underlying scope of this algorithm. We can apply the results of implementing this algorithm in the below stated ways.

Once we know the maximum likelihood of the expression of a gene from the estimated segment of the DNA, we can effectively try to mask that strand that expresses the gene by majority. This can be effective when trying to produce antibodies or other chemical compounds that can act as a mask by concealing the gene expression. Given the advantage that this is a machine learning algorithm that always produced the maximum probability of the path traversed by each base pair in the alignments so obtained, it may even be a little more precise in pointing out that DNA strand (and hence that chromosome in the example stipulated in this paper) that effectively expresses the gene.

VI. ACKNOWLEDGEMENTS

The author would like to thank her family and friends for unconditional support and motivation. She also thanks Suraj Rao K. P and Shreyas S Kaundinya, the author’s colleagues at PES University, as reviewers for their useful insights. The author is also immensely grateful for the constant mentoring and their comments on the earlier versions of this manuscript, although any errors are the author’s own and should not tarnish the reputations of the esteemed persons.

REFERENCES

[1] Katinka MD, Duprat S, Cornillot E, Méténier G, Thomarat F, Prensier G, Barbe V, Peyretaillade E, Brottier P, Wincker P, Delbac F, El Alaoui H, Peyret P, Saurin W, Gouy M, Weissenbach J, Vivarès CP. Genome sequence and gene compaction of the eukaryote parasite Encephalitozoon cuniculi. Nature. 2001 Nov 22.

[2] Wright, B.W., Molloy, M.P. & Jaschke, P.R. Overlapping genes in natural and engineered genomes. *Nat Rev Genet* **23**, 154–168 (2022)

[3] https://medium.com/analytics-vidhya/viterbi-algorithm-for-prediction-with-hmm-part-3-of-the-hmm-series-6466ce2f5dc6 (Resource: Viterbi Algorithm)

[4] Keeling PJ, Slamovits CH. Simplicity and complexity of microsporidian genomes. Eukaryot Cell. 2004 Dec;3(6):1363-9. doi: 10.1128/EC.3.6.1363-1369.2004. PMID: 15590811; PMCID: PMC539024.

[5] Abdelkhalik, O., and Darani, S. (June 4, 2018). "Evolving Hidden Genes in Genetic Algorithms for Systems Architecture Optimization." ASME. *J. Dyn. Sys., Meas., Control*. October 2018

[6] Bar-Joseph Z, Gerber G, Simon I, Gifford DK, Jaakkola TS. Comparing the continuous representation of time-series expression profiles to identify differentially expressed genes. Proc Natl Acad Sci U S A. 2003 Sep 2;100(18):10146-51. doi: 10.1073/pnas.1732547100. Epub 2003.

[7] Friedman N, Linial M, Nachman I, Pe'er D. Using Bayesian networks to analyze expression data. J Comput Biol. 2000

[8] Alexander Schliep, Alexander Schönhuth, Christine Steinhoff, Using hidden Markov models to analyze gene expression time course data, *Bioinformatics*, Volume 19, Issue suppl_1, 3 July 2003.