Implementing an HMM for Promoter Finding

Vishnu Gorur — Enes Kelestemur — Vien Vuong — Matthew Jakimovski
https://github.com/vienv2/cs466-final-project

Introduction

Gene finding is a very useful tool in creating genome annotations. Gene annotation files provide a lot of information when dealing with newly discovered genomes when there is no prior knowledge about the protein-coding genes. Computer methods for gene finding have been developed more than 20 years ago. One such program is Genemark.hmm, which is an intricate program to find genes in Prokaryotic genomes, however, it was too complex to implement. Furthermore, another computational method is CpG islands. CpG represents a dinucleotide that is very rare in the genome, however, this dinucleotide appears frequently around the promoter regions for many genes. As a result, this phenomenon is called CpG islands. The major benefit of using CpG islands is due to the nature of the complexity of the Eukaryotic genome. Gene finding in a Eukaryotic genome is extremely complicated due to the presence of exons and introns.

There are two main implementations of CpG islands, the first one is a purely algorithm-based method of using a sliding window to traverse the genome and find the protein-coding regions. However, this approach does have a pitfall with respect to fixing the size of the window since if the window is too short/long then it will not be able to accurately determine if it is in a protein-coding region or not in the protein-coding region. The alternative implementation is to use a Hidden Markov Model (HMM) to find CpG islands in a long sequence. The HMM has two hidden states, inside a CpG island and outside of the CpG island. These states emit symbols: A, C, G, T.

However, CpG islands exist only in Eukaryotic genomes and not in Prokaryotic genomes. The question we wanted to answer was, how can we find the states in a long sequence? We decided to use a different set of hidden states in our HMM implementation. In a prokaryotic genome, upstream of a promoter for a gene, specifically at the -35 and -10 locations, there are two key consensus sequences that can be used as a flag to identify nearby protein-coding sequences. Theoretically, both of these sequences should appear together since they are both needed for translation of the gene. Thus, our simple HMM model consisted of 3 hidden states: outside of both sequences, inside the -35 sequence, inside the -10 sequence. After performing a traceback on the matrix generated from the dynamic programming Viterbi algorithm, the optimal path can be used to find the location of the sequences. The parts that pass through the -10 and -35 states will be the sequences of interest.

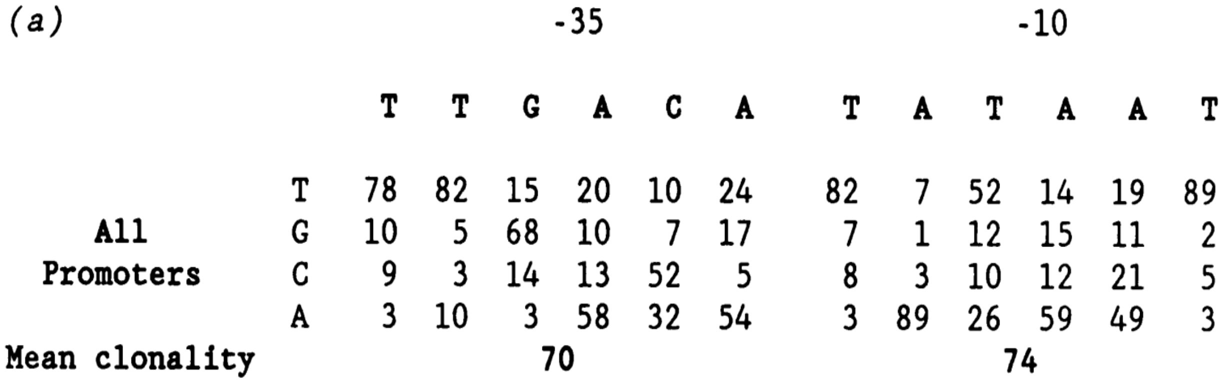
The specific Prokaryotic genome that was used in this implementation was the Escherichia coli str. K-12 substr. MG1655 genome. The dataset was obtained from the NCBI genome database. Using this sequence, and calculated probabilities, we were able to generate a traceback for the most probable path. However, one major improvement that could be made to our project would be to train our HMM model on training sets of sequences since the transition and emission probability matrices would be more realistic and accurate.

Methods

The project consists of three different components: determining states in HMM model, constructing probability functions (matrices), and HMM model. As any other HMM model structure, before implementing the HMM algorithm, we need to find the well-defined states and the probability functions. Both of these steps require an in-depth literature review of the transcriptional factors in bacteria. Prokaryotic cells have a less complicated transcription process compared to Eukaryotic cells. This relative simplicity allows us to detect hidden states that are potential indications of coding regions. After literature review, it was decided that there are four good options to be chosen as states. These states are -35 sequence, -10 sequence, start codon, and stop codon. -35 sequence is a hexamer that is seen approximately 35 bases upstream of the start codon. -10 is also a hexamer, and it is seen 10 bases upstream of the start codon. The start codon is where the coding region begins, and the stop codon is where the coding region ends.

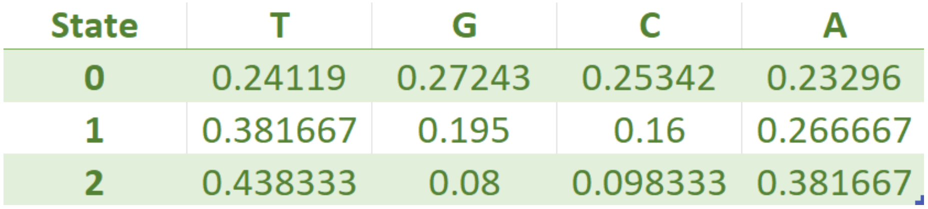
At first glance, these states might seem reasonable to use, but the start and stop codons are very short sequences. Moreover, using these sequences can be very problematic because there is a chance that they randomly occur in the genome. Therefore, we chose the -35 and -10 sequences to use as our states. This results in three different hidden states in the HMM model: outside of both states (state 0), inside of the -35 states (state 1), and inside -10 states (state 2).

Once the hidden states are decided, the next step is to understand the structure of probability functions. The motivation behind this project is to find the transcriptionally meaningful regions in newly discovered genomes. One of the most important parts is to construct a consensus sequence that represents the majority of the -10 and -35 sequences. Using consensus will allow us to identify promoter regions even if there is no historical knowledge about the new genome. However, it is important to understand that this assumption is hard to accept in some circumstances. For example, a newly discovered genome may not have a close hexamer region to the consensus sequence. For this project, to simplify the work, we build upon this assumption. In order to find the consensus sequence, we used literature, and table 1 was found for 112 promoters in E. coli (Harley). Since E. coli is the most widely studied organism, it was easy to find a promoter analysis for it.



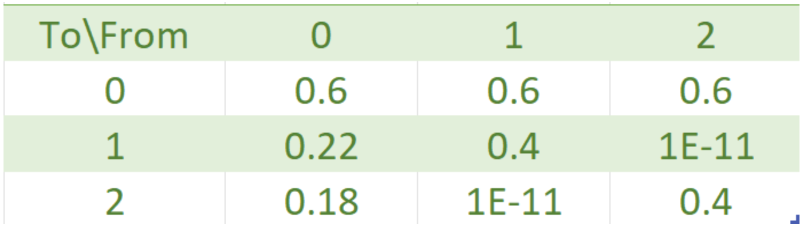
(a)				-35						-10			
		T	T	G	A	C	A	T	A	T	A	A	T
All Promoters	T	78	82	15	20	10	24	82	7	52	14	19	89
	G	10	5	68	10	7	17	7	1	12	15	11	2
	C	9	3	14	13	52	5	8	3	10	12	21	5
	A	3	10	3	58	32	54	3	89	26	59	49	3
Mean clonality				70						74			

The values in the table show that these sequences are somewhat fixed and introduce a consensus sequence. Moreover, the average of these values was used to calculate the emission probabilities for these two hidden states. To calculate the emission values for state 0, we counted the occurrences of each base and divided them by the complete genome size. As expected, the values are very close to 0.25 for each base. This calculation was not necessary considering the ultimate goal of the project which is to work on newly synthesized genomes, but for the sake of simplicity, we exclusively worked on the E. coli genome in this project. The final version of the emission matrix is given below.



State	T	G	C	A
0	0.24119	0.27243	0.25342	0.23296
1	0.381667	0.195	0.16	0.266667
2	0.438333	0.08	0.098333	0.381667

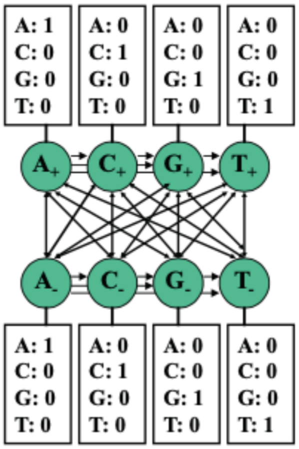
The calculation of the transition matrix was complicated. The first try was to identify the number of promoter regions and use that number to calculate the transition probabilities. However, this try failed because the number of promoter regions was very small compared to the whole genome. Therefore, the first probabilities were calculated very small. This gave us almost no transition between states and the model remained in state 0 throughout the iteration. The transition probabilities from 0 to other states were calculated by tuning these values. A reasonable range of transition probabilities was chosen, and iterating through the range, the optimal probabilities were decided. At the end of this tuning process, we checked the testing data to test whether the model correctly predicted the location of promoters or not. Finally, the optimal values are found for the transition probabilities. The final version of the matrix is given below.

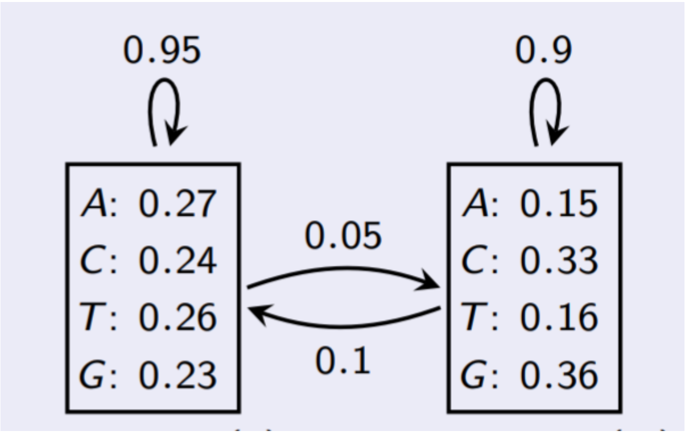


To\From	0	1	2
0	0.6	0.6	0.6
1	0.22	0.4	1E-11
2	0.18	1E-11	0.4

Lastly, the initial probabilities were calculated. These values were relatively easy to determine. As mentioned above, the promoters are very small compared to the whole genome which makes it very unlikely that the initial position will be inside of state 1 or 2. Therefore, it was determined that the initial value for state 0 was 1 and the other two states were 0.

The last component of the project is the construction of the HMM model. In the literature review, it was seen that there are two possible structures for the HMM model. The below figures illustrate these two structures (5 and 7). The figures are drawn for two hidden states. The primary difference between these two models is that one model splits the emission and transition probabilities while the other one combines these two. The figure on the right illustrates the model we used. It assumes that the promoter regions are the hidden states, and each state has its own emission probabilities. There are also transition probabilities between the states. This is more conventional and less complicated than the other model. The other structure (left) assumes that each base will be used as a state, so if there are two states initially, there will be 8 states since each base will have different values in different states. Therefore, this model does not have emission values (1 and 0s), and all the probabilistic information contained in the transition matrix. This model is more complicated than the conventional model, but it was discussed that it is a better model to detect promoter regions (5). The downside of this model is that the number of hidden states grows exponentially if we want to incorporate more states. For example, in our model we include 3 states, and this would make a 12 by 12 matrix for all of the transition probabilities.





All in all the model on the right was chosen for the sake of simplicity and computational complexity. The other aspect of the model is the HMM problem we are trying to solve. Since the purpose of this project is to find the promoter locations, we implemented the Viterbi algorithm. The results will provide a path of states for a given sequence. The code was tested on a snippet from the E. coli genome.

Validation Results

```
Sequence:    CTGTGTGGATTAAAATTGACAAAAGAGTGTCTGATAGCTATAATAGCTTCTGAAC
Our result:  --------+++++++++---------------------++++++-----------
Expected:    ---------------++++++-----------------++++++-----------
```

Figure 1: Observed recognition of the -35 and -10 sites compared to expected results

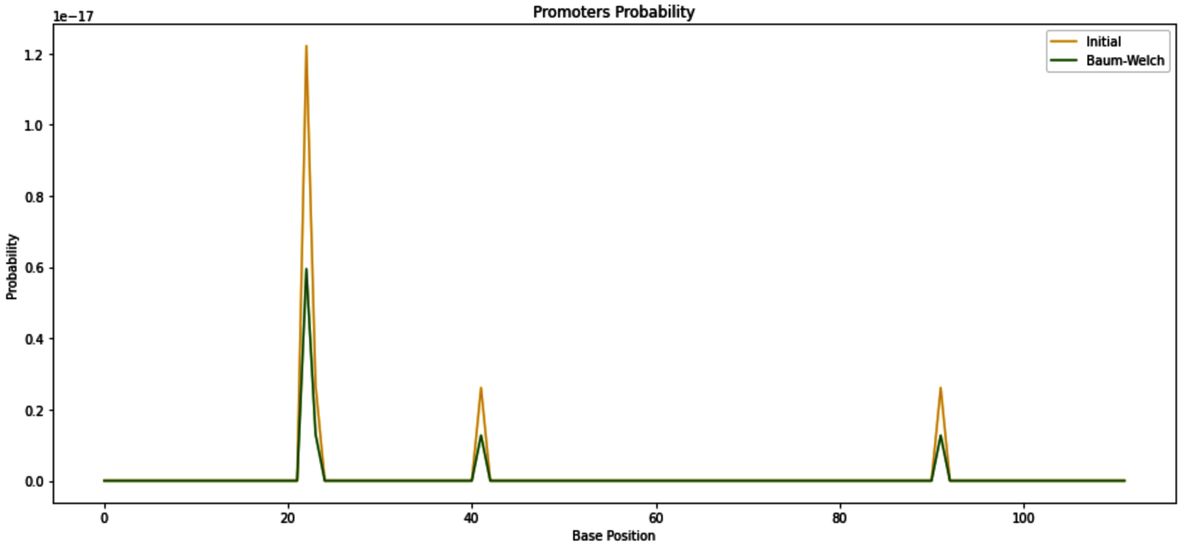


Figure 2: Promoter Probability vs Base Position for initial vs Trained HMM (Using Baum-Welch)

Conclusion + Future Work

The goal of this project was to identify the -35 and -10 sequence sites in a long sequence. Using a HMM we used hidden states to represent certain properties of the Ecoli genome. State 0 represented the sequence of nucleotides outside both sites, state 1 represented that we were in the -35 sequence site, state 2 represented that we were in the -10 site. Using this, we were able to predict the crucial sites that are indicative of a gene promoter downstream of the -10 site.

Although our results found the same -10 site as the expected result, the observed -35 sequence site was not matching up with the expected values. One possibility of why this didn't work could be due to emission probabilities. In the matrix, the emission probabilities for state 1 is closer to state 0 implying that these values are not too marginalized. However, in the same matrix we can see that the emission values for state 2 are quite marginalized given that there is no GC content in the consensus sequence TATAAT.

Taken together, our results indicate that our approach of using a HMM to find these sites on a prokaryotic genome is valid and can be used to find the location of gene promoters. However, we also learned that trying to hand calculate the transition and emission probabilities leads to slightly different results than expected. A solution to this is to use the Baum-Welch algorithm to have our HMM learn by expectation maximization from training sets of large amounts of sequence data so that it can generate the probabilities itself. This simple HMM can be built upon with additional hidden states such as factoring the stop codon for these genes etc, to get closer to a true biological representation of a prokaryotic genome, and be used in more complex gene-finding tasks.

References

1. Fickett,J.W. (1981) Recognition of protein coding regions in DNA sequences. Nucleic Acids Res., 10, 5303–5318.
2. Gribskov,M., Devereux,J. and Burgess,R.R. (1984) The codon preference plot: graphic analysis of protein coding sequences and prediction of gene expression. Nucleic Acids Res., 12, 539–549
3. Staden,R. (1984) Measurements of the effects that coding for a protein has on a DNA sequence and their use for finding genes. Nucleic Acids Res., 12, 551–567.
4. Santos-Zavaleta A et al. (2019). RegulonDB v 10.5: tackling challenges to unify classic and high throughput knowledge of gene regulation in E. coli K-12., *Nucleic Acids Res. 2019 Jan 8;47*(D1):D212-D220. doi: 10.1093/nar/gky1077
5. Harley, C. B., & Reynolds, R. P. (1987). Analysis of E. coli promoter sequences. *Nucleic acids research*, *15*(5), 2343–2361. https://doi.org/10.1093/nar/15.5.2343 Encoding Memory in a HMM- Detection of CpG islands. (2021, January 3). Massachusetts Institute of Technology. https://bio.libretexts.org/@go/page/40962
6. Blattner, F. R., Plunkett, G., 3rd, Bloch, C. A., Perna, N. T., Burland, V., Riley, M., Collado-Vides, J., Glasner, J. D., Rode, C. K., Mayhew, G. F., Gregor, J., Davis, N. W., Kirkpatrick, H. A., Goeden, M. A., Rose, D. J., Mau, B., & Shao, Y. (1997). The complete genome sequence of Escherichia coli K-12. *Science (New York,N.Y.)*, *277*(5331), 1453–1462. https://doi.org/10.1126/science.277.5331.1453
7. http://www.math.clemson.edu/~macaule/classes/f16_math4500/slides/f16_math4500_cpg-islands_handout.pdf

Member Information

Vishnu Gorur - vgorur2 - vgorur2@illinois.edu
Enes Kelestemur - enesk2 - enesk2@illinois.edu
Vien Vuong - vienv2 - vienv2@illinois.edu
Matthew Jakimovski - mjakim2 - mjakim2@illinois.edu