GENE FINDING WITH
GNENMARK.HMM

BY LUKASHIN & BORODOVSKY
What is Gene?

- DNA contains genetic instructions used in the development and functioning of all known living organisms and some viruses.

- A segment of DNA carrying the genetic information is called ‘Gene’.
  - Human has 3 billion base-pairs, but gene is typically few hundred base-pairs.
What is Gene?

- **Central Dogma**
  - Information flow from DNA(Gene) to mRNA to Protein
GeneMark.hmm

- Given DNA sequence $S = \{b_1, b_2, \ldots, b_L\}$, $b_i \in \{A,C,G,T\}$
- Find “functional sequence” $A = \{a_1, a_2, \ldots, a_L\}$
  - $a_i = 0$ if non-coding sequence,
  - $a_i = 1$ if coding sequence in forward strand
  - $a_i = 2$ if coding sequence in reverse strand

$S = \ldots$CGTAGTAGTAGCGGCGTAGAGATGTAG\ldots$
$A = \ldots$001111122012211101001221111000\ldots
GeneMark.hmm

Diagram:

- Direct start codon
- Non-coding state of length n nt
- Reverse stop codon
- Reverse strand coding state: Atypical gene of length m nt
- Reverse strand coding state: Typical gene of length k nt
- Direct strand coding state: Atypical gene of length j nt
- Direct strand coding state: Typical gene of length i nt
- Direct stop codon
GeneMark.hmm

- **HMM**
  - Non-coding state, Direct/Reverse strand coding state
    - choose
      - a length of sequence to emit
      - the sequence to emit
  - Typical/Atypical gene states
    - both emit coding sequence
    - but, different codon usage patterns
GeneMark.hmm

- **Hidden State Trajectory ‘A’**
  - Sequence of M hidden states \(a_i\) having duration \(d_i\)
    - \(A = \{(a_1, d_1), (a_2, d_2), \ldots, (a_M, d_M)\}\)
    - \(\sum d_i = L\)
  - Find \(A^*\) maximizing \(Pr(A \mid S)\)

\[
P_{\text{max}} = P(A^*, S) = \max_{(a_1d_1)(a_2d_2)\ldots(a_Md_M)} \Pr[(a_1d_1)(a_2d_2)\ldots(a_Md_M), b_1, b_2 \ldots b_L] \\
\sum_{s=1}^{M} d_s = L
\]

- \(A^*\): highest probability of occurring simultaneously with sequence \(S\)
- Use Viterbi algorithm
**GeneMark.hmm**

- Define (dynamic programming)
  - Joint probability of a partial trajectory of $m$ states and a partial sequence of length $l$

\[
Z_1(a_m, d_m) = \max_{(a_1 d_1) \ldots (a_{m-1} d_{m-1})} \sum_{d_1 = 1}^{m-1} d_s = l - d_m \left[ \text{Prob}\{a_1 d_1 \ldots a_{m-1} d_{m-1}, b_1 \ldots b_{l-d_m}\} q_{a_{m-1} a_m} \right]^{2} p_m(d_m) p_m(b_{l-d_m+1} \ldots b_l)
\]

- Transition probability
- Probability of duration
- Probability of sequence
\[ z_l(a_m, d_m) = \max_{(a_{m-1}, d_{m-1})} \left[ z_{l-1}(a_{m-1}, d_{m-1}) q a_{m-1} a_m \right] p a_m(d_m) p a_m(b_{l-d_m+1} \ldots b_L) \quad (3) \]

\[ \{ a^*_l(a), d^*_l(a) \} = \arg \max_{(a_m, d_m)} [z_l(a_m, d_m) q a_m a] \quad 2 \leq l \leq L-1 \quad (4) \]

\[ p_{\max} = \max_{(a_M, d_M)} z_L(a_M, d_M) \quad (5) \]

\[ \{ a^*_L, d^*_L \} = \arg \max_{(a_M, d_M)} [z_L(a_M, d_M)] \quad (6) \]
**GeneMark.hmm**

- **Parameters of HMM**
  - **Emission probabilities**
    - Coding and non-coding sequence
      - From previous study
    - Duration of sequence
      - From frequency distribution of lengths of known coding sequences
    - Start codon
      - \( \text{Pr(ATG)} = 0.905, \text{Pr(GTG)} = 0.090, \text{Pr(TTG)} = 0.005 \)
  - **Transition probabilities**
    - Non-coding to typical = 0.85
    - Non-coding to atypical = 0.15
**GeneMark.hmm**

- **Post-processing**
  - Gene may overlap, but HMM cannot capture this.
    - Viterbi algorithm tends to predict genes shorter than they really are.

  ![Gene Representation](image)

- **Find Ribosome binding site (RBS)**
  - If Score of RBS candidate > threshold, redefine initiation position (generating longest gene sequence)

*Ribosome binds and initiate protein translation*
Search for RBS
- Take 325 genes from E. coli (bacterium) with known RBS
- Align them using sequence alignment
- Use this as a PWM to scan for RBS
GeneMark.hmm

- **Data set**
  - Data set #1: all annotated E. coli genes
  - Data set #2: non-overlapping genes
  - Data set #3: Genes with known RBS
  - Data set #4: Genes with known start positions
## Result

### The GeneMark.hmm performance

<table>
<thead>
<tr>
<th>Set #</th>
<th>Number of genes</th>
<th>Prediction method</th>
<th>Exact prediction</th>
<th>Only 3'-end prediction</th>
<th>Missing genes</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>4288</td>
<td>VA</td>
<td>2483 (58%)</td>
<td>1592 (37%)</td>
<td>213 (5%)</td>
</tr>
<tr>
<td>1</td>
<td>4288</td>
<td>PP</td>
<td>3233 (75%)</td>
<td>842 (20%)</td>
<td>213 (5%)</td>
</tr>
<tr>
<td>2</td>
<td>2821</td>
<td>VA</td>
<td>2017 (71%)</td>
<td>750 (27%)</td>
<td>54 (2%)</td>
</tr>
<tr>
<td>2</td>
<td>2821</td>
<td>PP</td>
<td>2268 (80%)</td>
<td>499 (18%)</td>
<td>54 (2%)</td>
</tr>
<tr>
<td>3</td>
<td>325</td>
<td>VA</td>
<td>255 (78%)</td>
<td>64 (20%)</td>
<td>6 (2%)</td>
</tr>
<tr>
<td>3</td>
<td>325</td>
<td>PP</td>
<td>289 (89%)</td>
<td>30 (9%)</td>
<td>6 (2%)</td>
</tr>
<tr>
<td>4</td>
<td>204</td>
<td>VA</td>
<td>156 (76.5%)</td>
<td>47 (23%)</td>
<td>1 (0.5%)</td>
</tr>
<tr>
<td>4</td>
<td>204</td>
<td>PP</td>
<td>177 (87.5%)</td>
<td>26 (12%)</td>
<td>1 (0.5%)</td>
</tr>
</tbody>
</table>

VA: Viterbi algorithm  
PP: With post-processing
Result

- Gene overlap is important
  - Accuracy drops from 71% to 58% as excluding overlapping genes

- Post-processing is useful
  - Accuracy increases from 58% to 75% (data #1)

- Missing genes < 5%
  - Genes annotated, but not correctly predicted

- “Wrong” gene predictions ~8%
  - Genes predicted, but not match to any annotated gene
## Result

<table>
<thead>
<tr>
<th>Number of genes</th>
<th>Prediction method</th>
<th>Exact prediction</th>
<th>Only 3'-end prediction</th>
<th>Missing genes</th>
</tr>
</thead>
<tbody>
<tr>
<td>148</td>
<td>GeneMark.hmm</td>
<td>105 (71%)</td>
<td>28 (19%)</td>
<td>15 (10%)</td>
</tr>
<tr>
<td>148</td>
<td>GeneMark</td>
<td>92 (62%)</td>
<td>37 (25%)</td>
<td>19 (13%)</td>
</tr>
<tr>
<td>148</td>
<td>ECOPARSE</td>
<td>79 (53%)</td>
<td>33 (23%)</td>
<td>36 (24%)</td>
</tr>
</tbody>
</table>
Thanks!