CpG Island Modeling Using Graphical Models

Gang Ji, Tim Hg
Dept. Electrical Engineering

Lingyun Huang
Dept. Bioengineering
CpG Island

CpG island

- Short stretch in DNA with higher frequency of CG sequence

- Located around the promoter of house keeping Genes or other genes frequently expressed in a cell

- Due to different methylation level in inactive and active genes
Methylated cytosine

Not recognized by maintenance methylase

DNA replication

Cytosine

Deamination

5-methylcytosine

Thymine

Uracil

Deamination
CpG Island Modeling

**Hidden Markov Models**
- States: \( A^+, C^+, G^+, T^+ \)
- Observations: \( A, C, G, T \)
HMMs for CpG

- HMMs are good. But…
  - Conditional independent statements too strong
  - $X_t \perp X_i \mid S_t$

![Diagram of HMMs for CpG]
HMMs for CpG

- HMMs are good. But...
  - Duration Modeling
    - State occupancy decreases exponentially with time: \( d_i(t) = a_{ii}^t(1 - a_{ii}) \) \( \rightarrow \) poor duration modeling
  - Conditional independent statements too strong
    \[ X_t \perp X_i \mid S_t \]
    - Hard to effectively handle non-stationary observations that are highly correlated.
Our Proposed Improvements

- Language models
- Change the structure of graph
- Other graphical families (MRFs)
Topology of the HMMs

Two HMMs were used:
- \( N \): non-island
- \( Y \): island

Strictly Left-to-Right HMMs:

![Diagram of HMMs]

- Null States
- Emitting States

\( N \): Non-island

\( Y \): Island
HMM Training using HTK

Training Data:
- Discrete Observations: Mapped in indices
  - Discrete HMMs
- With Model Alignment:
  - Performed Baum-Welch training within the model:
    - Since only the state sequences are hidden
Decoding using HMM and Language Model

the Cost Function:

\[
\hat{W} = \arg \max_W P(W \mid O) = \arg \max_W \frac{P(W)P(O \mid W)}{P(O)} = \arg \max_W P(W)P(O \mid W) = \arg \max_W \left[ \log P(W) + \log P(O \mid W) \right]
\]
Issues

- \( P(O|W) \) is usually underestimated due to the fallacy of the Markov and independence assumptions. \( \rightarrow \) give the language model too little weight.

- Introduce language model weight \((LW)\) to balance the two probability quantities.

  Usually \( LW > 1.0 \) and it is task dependent

- The Cost Function becomes:

\[
\hat{W} = \arg\max_{W} [LW \cdot \log P(W) + \log P(O|W)]
\]
Decoding Framework

- No language model
  - Assuming all sequences are equally likely

WP: word penalty to compensate HMM prob.
Decoding with bigram LM

\[ P(W) \equiv P(w_1) \cdot P(w_2|w_1) \cdot P(w_3|w_2) \cdots P(w_n|w_{n-1}) \]
Evaluation Corpus

- Gene sequence
  - EMBL, European Bioinformatics Institute

- CpG island alignment
  - European Bioinformatics Institute

- We used
  - Whole corpus: 1710 sq.
  - Training: 1539 sq.
  - Testing: 171 sq.
## Corpus Statistics

<table>
<thead>
<tr>
<th></th>
<th>CpG island subsequence</th>
<th>DNA sequence</th>
</tr>
</thead>
<tbody>
<tr>
<td>Maximum</td>
<td>3340</td>
<td>185775</td>
</tr>
<tr>
<td>Minimum</td>
<td>181</td>
<td>44</td>
</tr>
<tr>
<td>Mean value</td>
<td>465</td>
<td>3787</td>
</tr>
</tbody>
</table>
Evaluation Metric

- No standard quantitative metric

- Precision/Recall
  - Precision
    - $P$: True positive / all hypothesized truth
  - Recall
    - $R$: True positive / all truth
Evaluation Metric

- No standard quantitative metric
- Precision/Recall
  - Precision
    \[ P: \text{True positive / all hypothesized truth} \]
  - Recall
    \[ R: \text{True positive / all truth} \]
  - F score (when no free parameter)
    \[ \frac{1}{F} = \frac{1}{P} + \frac{1}{R} \]
## Language Model Results

<table>
<thead>
<tr>
<th></th>
<th>Precision</th>
<th>Recall</th>
<th>F Measure</th>
</tr>
</thead>
<tbody>
<tr>
<td>Baseline</td>
<td>29.5%</td>
<td>77.7%</td>
<td>0.214</td>
</tr>
<tr>
<td>LM bigram</td>
<td>36.3%</td>
<td>75.0%</td>
<td>0.245</td>
</tr>
</tbody>
</table>
Graphical Models

- Nodes: random variables
- Edges: encodes conditional independent statements

earthquake → alarm
burglar → alarm

\[ P(a | e) \]
\[ P(a | b) \]
\[ P(e) \]
\[ P(b) \]
Graphical Models

- Different graphical models
  - Directed: Bayesian networks
  - Undirected: Markov random fields
  - Mixture of the two

- Next work
  - Dynamic Bayesian networks (DBNs)
  - Conditional random fields (CRFs)
DBNs

Dynamics Bayesian networks

- Directed graphical model
- Prologue/chunk/epilogue
- Unroll to fit series
- HMM is a DBN
Our DBN Models

Recall

– HMM CI statements too strong
– Idea: add dependencies in gene sequences
– 8 hidden states
Training

Standard EM learning

- island
- state
- gene
Decoding

- Junction tree algorithm
  - Form junction tree from the graph
  - Message passing along the tree
  - Viterbi assumption
ROC Curves

- Receiver operating characteristic curves
  - Free parameter to tune between precision and recall
DBN Results

![Graph showing DBN Results with sensitivity on the y-axis and 1-specificity on the x-axis, with lines representing different models: gene unigram, gene bigram, and gene trigram.](image)
## DBN Conclusions

- Adding links between gene observations helps a lot
- Equal error rates

<table>
<thead>
<tr>
<th>model</th>
<th>EER</th>
<th>rel. imp.</th>
</tr>
</thead>
<tbody>
<tr>
<td>baseline</td>
<td>38.8%</td>
<td>-</td>
</tr>
<tr>
<td>bigram</td>
<td>25.5%</td>
<td>34.3%</td>
</tr>
<tr>
<td>trigram</td>
<td>22.4%</td>
<td>42.3%</td>
</tr>
</tbody>
</table>
Conditional random field for labeling sequence

- An undirected acyclic graph
- Random field

Definition: for X is a random variable over observation sequence and Y is a random variable over state sequence.

\[(X, Y)\] forms a conditional random field

Laffetry et al. 2001
Conditional random field (CRF) example

- Comparison between CRF and HMM

HMM

CRF

\[ P(Y_3 \mid X, \text{all other } Y) = P(Y_3 \mid X, Y_2, Y_4) \]

Think of \( X \) as observations and \( Y \) as labels

Lafferty et al. 2001
Probabilistic Models of CRF

- Local features of CRF is specified by a vector \( f \) including
  - state feature
  - transition feature

Global feature \( F(y, x) \)

Conditional probability distribution defined by the CRF

\[
p_{\lambda}(Y|X) = \frac{\exp \lambda \cdot F(Y, X)}{Z_{\lambda}(X)}
\]

where

\[
Z_{\lambda}(x) = \sum_{y} \exp \lambda \cdot F(y, x)
\]
Decoding by CRF

The most probable label sequence for input sequence \( x \) is

\[
\hat{y} = \arg \max_y p_\lambda(y|x) = \arg \max_y \lambda \cdot F(y, x)
\]

The algorithm is also Viterbi

Training of CRF

– Generalized iterative scaling

given training set \( T = \{(x_k, y_k)\}_{k=1}^N \), which we assume fixed for the rest of this section:

\[
\mathcal{L}_\lambda = \sum_k \log p_\lambda(y_k|x_k) = \sum_k [\lambda \cdot F(y_k, x_k) - \log Z_\lambda(x_k)]
\]

To perform this optimization, we seek the zero of the gradient

\[
\nabla \mathcal{L}_\lambda = \sum_k \left[ F(y_k, x_k) - E_{p_\lambda(Y|x_k)} F(Y, x_k) \right]
\]

Fei Sha et.al 2003
In the project

- Training data
  - Long sequence was truncated every 100 bits to get non-CpG island or CpG island sub-sequences labeled with 1 (non-CpG island) and 2 (CpG island) respectively.

- Testing data
  - The whole sequence as input
  - Truncated sub-sequences as input
Software

– A CRF toolkit in Java from http://crf.sourceforge.net by Dr. Sunita Sarawagi in IIT Bombay

Result

– Disappointed, it DID NOT pick up any CpG island

The possible reason

– Truncated strategy does not fit the tool
– Unfamiliar with the source code