Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data

John Lafferty
Andrew McCallum
Fernando Pereira

WhizBang! Labs–Research
School of Computer Science, Carnegie Mellon University
Department of Computer and Information Science, University of Pennsylvania

presented by Ryanne Dolan
Applications

- computational biology, bio-informatics
  - analyze DNA/RNA
- computational linguistics
  - topic segmentation
  - part-of-speech tagging
  - information extraction
  - syntax disambiguation
- web search, web analytics
Classic Methods

- Hidden Markov Models (HMMs)
- Maximum Entropy Markov Models (MEMMs)
HMMs

- generative model - produce observations based on current state
- trained to produce observations typical of training sequences
- conditional, non-generative model
  - produce probability distribution of possible next states based on current state and observation
  - trained to predict next states given observations
Generative vs Non-Generative
MEMMs outperform HMMs in many tasks
- increased recall
- much improved precision

MEMMs can integrate observations at multiple levels, ex. letters, words, lines, paragraphs (called “features”)
MEMMs generate probability distribution of possible next states *given* current state
- transitions leaving a state compete against each other, but not among *all states*
- per-state normalization of probability distribution (sum to 1.0)
- probability biased towards states with few transitions
- demonstrated experimentally
Label Bias Problem

- training sequences:
  - A B C D
  - A B D D
  - A B C E
  - A B D C
  - A B D C

- model says:
  - C -> D 50%
  - C -> E 50%

- why predict E when D is much more common in training sequences?
Proposed Solution

- model probability of transitions and probability of states
- Conditional Random Fields:
  - models probability of transitions between states
  - probability is conditional on current observation
  - not normalized – very different from HEMMs
  - considers many “features” of observations
CRFs use a set of predefined “edge features” and “vertex features”
- if word is capitalized and label is “proper noun”
- if word begins with number
- if word contains hyphen
- if word ends in “-ing”
- features are real power of CRFs
CRFs

- graph $G = (V, E)$
- random variable $X$ over data sequences
- random variable $Y$ over labels for sequences
- features
CRFs

- probability of labeling 'x' with label 'y'

\[ p_\theta(y \mid x) \propto \exp \left( \sum_{e \in E, k} \lambda_k f_k(e, y \mid e, x) + \sum_{v \in V, k} \mu_k g_k(v, y \mid v, x) \right) \]

- train to learn parameters:

\[ \theta = (\lambda_1, \lambda_2, \ldots; \mu_1, \mu_2, \ldots) \]
Training

- given:
  - set of features
  - graph (possibly fully-connected) mapping observations to possible labels
  - training sequences

- find: \[ \theta = (\lambda_1, \lambda_2, \ldots; \mu_1, \mu_2, \ldots) \]
  - effect of each edge feature on a transition
  - effect of each vertex feature on a state
Improved Iterative Scaling method estimates maximal-likelihood parameters for exponential models

CRFs are exponential models

CRF training algorithms extend IIS
Training

$p_{\Lambda}(y \mid x) \neq \frac{1}{Z_{\Lambda}(x)} \exp \left( \sum_{i=1}^{n} \lambda_i f_i(x, y) \right)$

to normalized

$p_{\theta}(y \mid x) \propto \exp \left( \sum_{e \in E, k} \lambda_k f_k(e, y \mid e, x) + \sum_{v \in V, k} \mu_k g_k(v, y \mid v, x) \right)$

for each edge and vertex
## Results

<table>
<thead>
<tr>
<th>model</th>
<th>error</th>
<th>oov error</th>
</tr>
</thead>
<tbody>
<tr>
<td>HMM</td>
<td>5.69%</td>
<td>45.99%</td>
</tr>
<tr>
<td>MEMM</td>
<td>6.37%</td>
<td>54.61%</td>
</tr>
<tr>
<td>CRF</td>
<td>5.55%</td>
<td>48.05%</td>
</tr>
<tr>
<td>MEMM+</td>
<td>4.81%</td>
<td>26.99%</td>
</tr>
<tr>
<td>CRF+</td>
<td>4.27%</td>
<td>23.76%</td>
</tr>
</tbody>
</table>

+ Using spelling features

Sorry, no pictures....
Conclusions

- HMMs good when nothing is known about process (except assumed to be Markovian)
- MEMMs and CRFs can use predefined features to greatly improve performance
- CRFs outperform MEMMs in author's experiments
- CRFs guaranteed to converge to maximal-likelihood parameters using IIS methods