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

John Lafferty, Andrew McCallum, Fernando Pereira

Presentation by Rongkun Shen
Nov. 20, 2003
Sequence Segmenting and Labeling

• Goal: mark up sequences with content tags

• Application in computational biology
  – DNA and protein sequence alignment
  – Sequence homolog searching in databases
  – Protein secondary structure prediction
  – RNA secondary structure analysis

• Application in computational linguistics & computer science
  – Text and speech processing, including topic segmentation, part-of-speech (POS) tagging
  – Information extraction
  – Syntactic disambiguation
Example: Protein secondary structure prediction

Conf: 977621015677468999723631357600330223342057899861488356412238
Pred: CCCCCCCCCCCCCCEEEEEECCCCCCCCCCCHHHHHHHHHHHHHHHCCCCCCCCC
AA: EKKSINECDLGGKKVLIRVDFNVPKNGKITNDYRISALPLTLLKVLTEGGSCVLMH
    10     20     30     40     50     60

Conf: 85576422245412347898510001047899999874033445740023666631258
Pred: CCCCCCCCCCCCCCCCCCCCCCCCCCHHHHHHHHHHHHHHHHHHHCCCCCCCCC
AA: RPKGIPQAQAGKIRSTGGVGPGFQQKATLKPVAKRLSELLLRPVTFAPDCLNAADVVSKMS
    70     80     90    100    110    120

Conf: 8746886110023430431001789999987505335521224334552001322452
Pred: CCCCCCECCCCHHHHHHHHHHHHHHHHHHHHHHHHHCCCCCCCCC
AA: PGDVVLLENURFYKEEGSKKAKDREAMAKILASYGDVYISDAFGTAHRDSATMTGIPKL
    130    140    150    160    170    180
Generative Models

- Hidden Markov models (HMMs) and stochastic grammars
  - Assign a joint probability to paired observation and label sequences
  - The parameters typically trained to maximize the joint likelihood of train examples

\[ P(X, Y) = \prod_i P(X_i | Y_i) P(Y_i | Y_{i-1}) \]
Generative Models (cont’d)

- **Difficulties and disadvantages**
  - Need to enumerate all possible observation sequences
  - Not practical to represent multiple interacting features or long-range dependencies of the observations
  - Very strict independence assumptions on the observations
Conditional Models

- Conditional probability $P(\text{label sequence } y \mid \text{observation sequence } x)$ rather than joint probability $P(y, x)$
  - Specify the probability of possible label sequences given an observation sequence

- Allow arbitrary, non-independent features on the observation sequence $X$

- The probability of a transition between labels may depend on past and future observations
  - Relax strong independence assumptions in generative models
Discriminative Models
Maximum Entropy Markov Models (MEMMs)

- Exponential model
- Given training set $X$ with label sequence $Y$:
  - Train a model $\theta$ that maximizes $P(Y|X, \theta)$
  - For a new data sequence $x$, the predicted label $y$ maximizes $P(y|x, \theta)$
  - Notice the per-state normalization

\[
P(y' | y, x) = \frac{1}{Z(y, x)} \exp \left( \sum_k \lambda_k \frac{f_k(x, y, y')}{\text{weight} \text{feature}} \right)
\]

$A = a$, where $A$ is a random variable, and $a$ is an outcome

![Diagram of MEMMs](image.png)
MEMMs have all the advantages of Conditional Models

Per-state normalization: all the mass that arrives at a state must be distributed among the possible successor states ("conservation of score mass")

Subject to Label Bias Problem

- Bias toward states with fewer outgoing transitions
Label Bias Problem

• Consider this MEMM:

![MEMM Diagram]

• $P(1 \text{ and } 2 \mid ro) = P(2 \mid 1 \text{ and } ro)P(1 \mid ro) = P(2 \mid 1 \text{ and } o)P(1 \mid r)$
  $P(1 \text{ and } 2 \mid ri) = P(2 \mid 1 \text{ and } ri)P(1 \mid ri) = P(2 \mid 1 \text{ and } i)P(1 \mid r)$

• Since $P(2 \mid 1 \text{ and } x) = 1$ for all $x$, $P(1 \text{ and } 2 \mid ro) = P(1 \text{ and } 2 \mid ri)$
In the training data, label value 2 is the only label value observed after label value 1
Therefore $P(2 \mid 1) = 1$, so $P(2 \mid 1 \text{ and } x) = 1$ for all $x$

• However, we expect $P(1 \text{ and } 2 \mid ri)$ to be greater than $P(1 \text{ and } 2 \mid ro)$.

• Per-state normalization does not allow the required expectation
Solve the Label Bias Problem

• Change the state-transition structure of the model
  – Not always practical to change the set of states

• Start with a fully-connected model and let the training procedure figure out a good structure
  – Prelude the use of prior, which is very valuable (e.g. in information extraction)
Random Field

Let $G = (Y, E)$ be a graph where each vertex $Y_v$ is a random variable. Suppose $P(Y_v \mid \text{all other } Y) = P(Y_v \mid \text{neighbors}(Y_v))$ then $Y$ is a random field.

Example:

- $P(Y_5 \mid \text{all other } Y) = P(Y_5 \mid Y_4, Y_6)$
Conditional Random Fields (CRFs)

- CRFs have all the advantages of MEMMs without label bias problem
  - MEMM uses per-state exponential model for the conditional probabilities of next states given the current state
  - CRF has a single exponential model for the joint probability of the entire sequence of labels given the observation sequence
- Undirected acyclic graph
- Allow some transitions “vote” more strongly than others depending on the corresponding observations
Definition of CRFs

\( \mathbf{X} \) is a random variable over data sequences to be labeled
\( \mathbf{Y} \) is a random variable over corresponding label sequences

**Definition.** Let \( G = (V, E) \) be a graph such that \( \mathbf{Y} = (Y_v)_{v \in V} \), so that \( \mathbf{Y} \) is indexed by the vertices of \( G \). Then \( (\mathbf{X}, \mathbf{Y}) \) is a conditional random field in case, when conditioned on \( \mathbf{X} \), the random variables \( Y_v \) obey the Markov property with respect to the graph:
\[
p(Y_v | \mathbf{X}, Y_w, w \neq v) = p(Y_v | \mathbf{X}, Y_w, w \sim v),
\]
where \( w \sim v \) means that \( w \) and \( v \) are neighbors in \( G \).
Example of CRFs

Suppose $P(Y_\nu | X, \text{all other } Y) = P(Y_\nu | X, \text{neighbors}(Y_\nu))$
then $X$ with $Y$ is a \textbf{conditional} random field

- $P(Y_3 | X, \text{all other } Y) = P(Y_3 | X, Y_2, Y_4)$
- Think of $X$ as observations and $Y$ as labels
Graphical comparison among HMMs, MEMMs and CRFs

Figure 2. Graphical structures of simple HMMs (left), MEMMs (center), and the chain-structured case of CRFs (right) for sequences. An open circle indicates that the variable is not generated by the model.
Conditional Distribution

If the graph $G = (V, E)$ of $Y$ is a tree, the conditional distribution over the label sequence $Y = y$, given $X = x$, by fundamental theorem of random fields is:

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

$x$ is a data sequence
$y$ is a label sequence
$v$ is a vertex from vertex set $V = \text{set of label random variables}$
$e$ is an edge from edge set $E$ over $V$
$f_k$ and $g_k$ are given and fixed. $g_k$ is a Boolean vertex feature; $f_k$ is a Boolean edge feature
$k$ is the number of features
$\theta = (\lambda_1, \lambda_2, \ldots, \lambda_n; \mu_1, \mu_2, \ldots, \mu_n); \lambda_k$ and $\mu_k$ are parameters to be estimated
$y|_e$ is the set of components of $y$ defined by edge $e$
$y|_v$ is the set of components of $y$ defined by vertex $v$
Conditional Distribution (cont’d)

- CRFs use the observation-dependent normalization $Z(x)$ for the conditional distributions:

$$p_\theta(y \mid x) = \frac{1}{Z(x)} \exp \left( \sum_{e \in E, k} \lambda_k f_k (e, y \mid e, x) + \sum_{v \in V, k} \mu_k g_k (v, y \mid v, x) \right)$$

$Z(x)$ is a normalization over the data sequence $x$
Parameter Estimation for CRFs

- The paper provided iterative scaling algorithms
- It turns out to be very inefficient
- Prof. Dietterich’s group applied Gradient Descendent Algorithm, which is quite efficient
Training of CRFs (From Prof. Dietterich)

• First, we take the log of the equation

$$
\log p_{\theta}(y \mid x) = \sum_{e \in E, k} \lambda_k f_k(e, y_e, x) + \sum_{v \in V, k} \mu_k g_k(v, y_v, x) - \log Z(x)
$$

• Then, take the derivative of the above equation

$$
\frac{\partial \log p_{\theta}(y \mid x)}{\partial \theta} = \frac{\partial}{\partial \theta} \left( \sum_{e \in E, k} \lambda_k f_k(e, y_e, x) + \sum_{v \in V, k} \mu_k g_k(v, y_v, x) - \log Z(x) \right)
$$

• For training, the first 2 items are easy to get.
• For example, for each $\lambda_k f_k$ is a sequence of Boolean numbers, such as 00101110100111.
  $\lambda_k f_k(e, y_e, x)$ is just the total number of 1’s in the sequence.

• The hardest thing is how to calculate $Z(x)$
Training of CRFs (From Prof. Dietterich) (cont’d)

• Maximal cliques

\[ c_1 : \exp(\varphi(y_1,x) + \varphi(y_2,x) + \psi(y_1,y_2,x)) = c_1(y_1,y_2,x) \]
\[ c_2 : \exp(\varphi(y_3,x) + \psi(y_2,y_3,x)) = c_2(y_2,y_3,x) \]
\[ c_3 : \exp(\varphi(y_4,x) + \psi(y_3,y_4,x)) = c_3(y_3,y_4,x) \]

\[ Z(x) = \sum_{y_1,y_2,y_3,y_4} c_1(y_1,y_2,x)c_2(y_2,y_3,x)c_3(y_3,y_4,x) \]
\[ = \sum_{y_1} \sum_{y_2} c_1(y_1,y_2,x) \sum_{y_3} c_2(y_2,y_3,x) \sum_{y_4} c_3(y_3,y_4,x) \]
Modeling the label bias problem

- In a simple HMM, each state generates its designated symbol with probability $29/32$ and the other symbols with probability $1/32$

- Train MEMM and CRF with the same topologies

- A run consists of 2,000 training examples and 500 test examples, trained to convergence using Iterative Scaling algorithm

- CRF error is 4.6%, and MEMM error is 42%

- MEMM fails to discriminate between the two branches

- CRF solves label bias problem
MEMM vs. HMM

- The HMM outperforms the MEMM
MEMM vs. CRF

- CRF usually outperforms the MEMM
CRF vs. HMM

Each open square represents a data set with $\alpha < 1/2$, and a solid circle indicates a data set with $\alpha \geq 1/2$; When the data is mostly second order ($\alpha \geq 1/2$), the discriminatively trained CRF usually outperforms the HMM.
POS tagging Experiments

UPenn tagging task: 45 tags (syntactic), 1M words training

The asbestos fiber, crocidolite, is unusually resilient

once it enters the lungs; with even brief exposures
to it causing symptoms that show up decades later;

researchers said
POS tagging Experiments (cont’d)

• Compared HMMs, MEMMs, and CRFs on Penn treebank POS tagging
• Each word in a given input sentence must be labeled with one of 45 syntactic tags
• Add a small set of orthographic features: whether a spelling begins with a number or upper case letter, whether it contains a hyphen, and if it contains one of the following suffixes: -ing, -ogy, -ed, -s, -ly, -ion, -tion, -ity, -ies
• oov = out-of-vocabulary (not observed in the training set)

<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
Summary

• Discriminative models are prone to the label bias problem

• CRFs provide the benefits of discriminative models

• CRFs solve the label bias problem well, and demonstrate good performance
Thanks for your attention!

Special thanks to
Prof. Dietterich & Tadepalli!