Graphical Models for Sequence Labeling in NLP

Anup Kulkarni

Indian Institute of Technology, Bombay

Sep 30, 2009

Under the guidance of Prof. Pushpak Bhattacharyya.
1. Introduction

2. Sentence, Graph and Random Fields

3. Maximum Entropy Markov Model

4. Conditional Random Fields
Graphical Models and Machine Learning

- Machine Learning like SVM, NB, K-NN have attributes as input and class label as output
- Attributes influence class label but class label does not influence other class labels

- Graphical Model attributes as well as class labels influence class labels

- Text Processing, NLP
- Computer Vision eg. Scene Recognition, Object Labeling etc.
Goal of NLP

- Understanding Language

The heart operation was painful.

He plays violin in the play.
Goal of NLP

- Understanding Language

The heart operation was painful.

He plays violin in the play.
Goal of NLP

- Understanding Language

The heart operation was painful.

He plays violin in the play.
Goal of NLP

- Understanding Language
- Sequence Labeling

The heart operation was painful.

He plays violin in the play.
What Labels?

Parts of Speech Tags

The_AT postman_NN collected_VBD letters_NNS and_CNJ left_VBD. He_PN plays_VBZ violin_NN in_PP the_AT play_NN.
What Labels?

- Parts of Speech Tags: POS tags
- Chunk Tags

He\(< B \rightarrow NP >\) reckons\(< B \rightarrow VP >\) the\(< B \rightarrow NP >\) current\(< I \rightarrow NP >\) account\(< I \rightarrow NP >\) deficit\(< I \rightarrow NP >\) will\(< B \rightarrow VP >\) narrow\(< I \rightarrow VP >\) .\(< O >\)
I bought 100 shares of IBM Corp.<ORG>.
IIT Bombay<ORG> is located in Powai which is in north part of Bombay<LOC>.
Mr. Wesley<PER> went to buy the book published by Addison Wesley<ORG>. 
- Words influence each other.
- Words can be represented using nodes.
- Relation between words can be represented using edges.

**Figure:** Graphical model induced on a sentence.
Imposing Graph on Sentence

- Words influence each other.
- Words can be represented using nodes.
- Relation between words can be represented using edges.

**Figure:** Graphical model induced on a sentence.

**Markov Assumption** Every node depends only on previous node.
Labeling Each Node

**Figure:** Graphical model induced on a sentence.
Each node can take all possible labels

If we define $y_i$ as random variable for node $i$ taking some label, we will get random field for

$$ Y = (y_1, y_2, \ldots, y_n) $$

Probability of tag sequence is given by

$$ Pr(Y) = Pr(y_1, y_2, \ldots, y_n) $$

Our goal is to find best probable tag sequence, given sentence (or observation) $X = (x_1, x_2, \ldots, x_n)$

$$ Pr(Y|X) = Pr(y_1, y_2, \ldots, y_n|X) $$
Hammersley-Clifford Theorem

Every Markov Random Field is equivalent to Gibbs Random Field
Hammersley-Clifford Theorem

Every Markov Random Field is equivalent to Gibbs Random Field

Markov Random Field is random field satisfying Markov Property

\[ \Pr(y_i|y_{N-i}) = \Pr(y_i|\text{Ne}(i)) \]

Gibbs Random Field is random field obeying Gibbs distribution

\[ \Pr(y) = \frac{\exp(-U(y))}{Z} \]

Where \( Z \) is defined as

\[ Z = \sum_{y \in Y} \exp(-U(y)) \]

U(y) is called as energy function defined as

\[ U(y) = \sum_{c \in C} V_c(y) \]

That is sum of all clique potentials.
Let \( D_i = Ne(i) \cup X_i \) be set of neighbors of \( X_i \) and \( X_i \) itself.
We have to prove that if Gibbs distribution is satisfied then markovian property is hold. That means given Gibbs distribution we have to arrive at

\[
\Pr(X_i|X_{S-i}) = \Pr(X_i|X_{Ne(i)})
\] (1)
Proof of HC Theorem $\textit{GRF} \rightarrow \textit{MRF}$

Starting with LHS

$$\Pr(X_i|X_{S-i}) = \frac{\Pr(X_i, X_{S-i})}{\Pr(X_{S-i})}$$

$$= \frac{\Pr(X_i, X_{S-i})}{\sum_{X_i} \Pr(X_i, X_{S-i})}$$

$$= \frac{\exp(\sum_{c \in C} V_c(X))}{Z(X)}$$

$$= \frac{\exp(\sum_{c \in C} V_c(X))}{\sum_{X_i} \exp(\sum_{c \in C} V_c(X))}$$

$$= \frac{\exp(\sum_{c \in C} V_c(X))}{\sum_{X_i} \exp(\sum_{c \in C} V_c(X))}$$
Proof of HC Theorem $GRF \rightarrow MRF$

Now cliques $C$ in the graph is set of cliques $A$ which contains node $X_i$ and $B$ which do not contain node $X_i$. Using this,

$$\Pr(X_i|X_{S-i}) = \frac{\exp(\sum_{c \in C} V_c(X))}{\sum_{X_i} \exp(\sum_{c \in C} V_c(X))}$$

$$= \frac{\exp(\sum_{c \in A} V_c(X)) \times \exp(\sum_{c \in B} V_c(X))}{\sum_{X_i} \exp(\sum_{c \in A} V_c(X)) \times \exp(\sum_{c \in B} V_c(X))}$$

since cliques in $B$ do not contain node $X_i$, the term $\exp(\sum_{c \in B} V_c(X))$ can be brought out of summation.

$$\Pr(X_i|X_{S-i}) = \frac{\exp(\sum_{c \in A} V_c(X)) \times \exp(\sum_{c \in B} V_c(X))}{\sum_{X_i} \exp(\sum_{c \in A} V_c(X)) \times \exp(\sum_{c \in B} V_c(X))}$$

$$= \frac{\exp(\sum_{c \in A} V_c(X))}{\sum_{X_i} \exp(\sum_{c \in A} V_c(X))}$$

The term in the numerator contains all the cliques which contains $X_i$. 

Anup Kulkarni (IIT-B)  
GM for Sequence Labeling in NLP  
Sep 30, 2009
Proof of HC Theorem \( GRF \rightarrow MRF \)

We arrive at the same expression from RHS as

\[
\Pr(X_i | X_{Ne(i)}) = \frac{\Pr(X_i, X_{Ne(i)})}{\Pr(X_{Ne(i)})} = \frac{\exp(\sum_{c \in A} V_c(X))}{Z(X)} = \frac{\sum_{X_i} \exp(\sum_{c \in A} V_c(X))}{\sum_{X_i} \exp(\sum_{c \in A} V_c(X))}
\]

Thus \( GRF \rightarrow MRF \) is proved.
Proof of HC Theorem $MRF \rightarrow GRF$

For this Mobius inversion principle is used,

$$F(A) = \sum_{B:B \subseteq A} G(B) \iff G(B) = \sum_{B:B \subseteq A} (-1)^{|B|-|C|} F(C)$$

This when written for energy function $U(X)$ and potentials $V(X)$,

$$U(x_v) = \sum_{B:B \subseteq V} V(x_b) \iff V(x_b) = \sum_{C:C \subseteq B} (-1)^{|B|-|C|} U(x_c)$$

where $V$ our graph and $B$ is subgraph.
Gibbs distribution requires energy to be defined over all cliques of the graph. So to prove that

$$V(x_b) = 0$$

whenever $B$ is not completely connected i.e. $B$ is not clique.
when $B$ is not a clique, let $Z_1$ and $Z_2$ be two nodes such that they are not connected in $B$.
let $S$ be separator : $Z_1SZ2 = B$
Proof of HC Theorem \( \text{MRF} \to \text{GRF} \)

\[
V(x_b) = \sum_{C:C \subseteq B} (-1)^{|B|-|C|} U(x_c)
\]
can be written as

\[
V(x_b) = \sum_{C':C' \subseteq S} (-1)^{|B|-|C'|} U(x_{c'}) \\
+ \sum_{C':C' \subseteq S} (-1)^{|B|-|Z_1 C'|} U(x_{z_1 c'}) \\
+ \sum_{C':C' \subseteq S} (-1)^{|B|-|C' Z_2|} U(x_{c' z_2}) \\
+ \sum_{C':C' \subseteq S} (-1)^{|B|-|Z_1 C' Z_2|} U(x_{z_1 c' z_2})
\]

adjusting exponents of \((-1)\)

\[
V(x_b) = \sum_{C':C' \subseteq S} (-1)^{|B|-|C'|} [U(x_{c'}) - U(x_{z_1 c'}) - U(x_{c' z_2}) + U(x_{z_1 c' z_2})] \quad (3)
\]

Now we will prove that bracketed term \([.]\) is 0. Taking exp of \([.]\) term,
Proof of HC Theorem $MRF \rightarrow GRF$

\[\begin{align*}
\exp[.] & = \exp(U(x_{c'}) - U(x_{z1c'}) - U(x_{c'z2}) + U(x_{z1c'z2})) \\
& = \frac{\exp(U(x_{c'})) \times \exp(U(x_{z1c'z2}))}{\exp(U(x_{z1c'})) \times \exp(U(x_{c'z2}))} \\
& = \frac{\exp(U(x_{c'}))}{Z} \times \frac{Z}{Z} \times \frac{\exp(U(x_{z1c'z2}))}{Z} \\
& = \frac{\Pr(x_{c'})}{\Pr(x_{z1c'})} \times \frac{\Pr(x_{z1c'z2})}{\Pr(x_{c'z2})}
\end{align*}\]
Using Baye's Theorem,

\[
\exp[.] = \frac{\Pr(x_{z_1}|x_{c'z_2})}{\Pr(x_{z_1}|x_{c'})}
\]

since \(X_{Z_1}\) and \(X_{Z_2}\) are not directly connected, using Markovian property,

\[
\Pr(x_{z_1}|x_{c'z_2}) = \Pr(x_{z_1}|x_{c'})
\]

\[
\exp[.] = 1
\]

so \([.]\) in 3 is 0. So by 2 Every MRF is GRF.
1. Maximum Entropy Markov Model
2. Cyclic Dependency Network
3. Conditional Random Field
Entropy

Entropy is a measure of the uncertainty associated with a random variable.

\[ H(X) = - \sum_{i=1}^{n} \Pr(x_i) \log \Pr(x_i) \]

Maximum Entropy

When incomplete information is available the only unbiased assumption which can be made is maximize uncertainty, i.e. Maximize Entropy.
Drawbacks of traditional HMM

- characteristics of text are not used
- learns joint distribution $Pr(y, x)$
- assumes wrong model: word depends on tag
Drawbacks of traditional HMM

- characteristics of text are not used
- learns joint distribution $\Pr(y, x)$
- assumes wrong model: word depends on tag

$$\Pr(y, x) = \Pr(y) \Pr(y|x)$$

Figure: A. Dependency Graph for traditional HMM B. Dependency Graph for Maximum Entropy Markov Model (MEMM) Source: [McCallum et. al., 2000]
Entropy
Entropy is a measure of the uncertainty associated with a random variable.

\[ H(X) = - \sum_{i=1}^{n} \Pr(x_i) \log \Pr(x_i) \]

Maximum Entropy
When incomplete information is available the only unbiased assumption which can be made is maximize uncertainty, ie. Maximize Entropy
Entropy is a measure of the uncertainty associated with a random variable.

\[ H(X) = - \sum_{i=1}^{n} \Pr(x_i) \log \Pr(x_i) \]

Maximum Entropy

When incomplete information is available, the only unbiased assumption which can be made is to maximize uncertainty, i.e., Maximize Entropy.

Incomplete Information

\[ \sum_y \Pr(y|x) = 1 \]
Formulation of MEMM

Given: pairs of (obs,label): \((x_1, y_1), (x_2, y_2), ..., (x_n, y_n)\)
Formulation of MEMM

- Given: pairs of (obs,label): \((x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n)\)
- **Define** Empirical Distribution \(\Pr(\tilde{x}, y)\)

\[
\tilde{\Pr}(x, y) = \frac{\text{No. of times } (x,y) \text{ occurs in sample}}{N}
\]
Formulation of MEMM

- Given: pairs of (obs,label): $(x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n)$
- **Define** Empirical Distribution $\Pr^{\sim}(x, y)$

$$\Pr(x, y) = \frac{\text{No. of times } (x,y) \text{ occurs in sample}}{N}$$

**Features** Extra information and characteristics of the text are represented using features

$$f(x, y) = \begin{cases} 1 & \text{if } x = \text{play}, y = \text{verb} \\ 0 & \text{otherwise} \end{cases}$$
Constrain on MaxEnt model

Expectation of features in MaxEnt model should be equal to expectation of features in training data.

Expectation in Empirical Distribution

\[
\tilde{\Pr}(f) = \sum_{x,y} \tilde{\Pr}(x,y)f(x,y)
\]

Expectation in MaxEnt Model

\[
\Pr(f) = \sum_{x,y} \Pr(x,y)f(x,y)
\]

\(^a\)Using same notation as Berger et. al. 1996
Constrain on MaxEnt model

expectation of features in MaxEnt model should be equal to expectation of features in training data.

Expectation in Empirical Distribution

\[ E_{x \sim (f)} = \sum_{x,y} \Pr(x, y) f(x, y) \]

Expectation in MaxEnt Model

\[ E_x(f) = \sum_{x,y} \Pr(x, y) f(x, y) \]

\[ \sum_{x,y} \Pr(x, y) f(x, y) = \sum_{x,y} \Pr(x) \Pr(y|x) f(x, y) \] (4)
Using MaxEnt Principle

**Maximum Entropy Model** Model that will maximize the entropy

\[
\Pr(y|x) = \arg \max_{\Pr(y|x)} H(y|x)
\]

where,

\[
H(y|x) = - \sum_{(x,y)} \tilde{\Pr}(x) \Pr(y|x) \log \Pr(y|x)
\]

\[
\sum_{x,y} \tilde{\Pr}(x,y)f(x,y) = \sum_{x,y} \tilde{\Pr}(x) Pr(y|x)f(x,y)
\]

\[
\sum_y \Pr(y|x) = 1
\]
Formulating Dual Objective

Let $\lambda_1 \ldots m$ be Langrange Multiplier associated 4 for each feature $f_1 \ldots m$. Let $\lambda_{m+1}$ be the Langrange Multiplier associated 5. Then MaxEnt dual objective is:

$$D(\Pr(y|x), \lambda) = H(y|x) + \sum_{i=1}^{m} \lambda_i (\mathbb{E}(f_i) - \mathbb{E}^\sim(f_i)) + \lambda_{m+1} (\sum_y \Pr(y|x) - 1)$$

Differentiating and equating to zero we get MaxEnt probability model as,

$$\Pr(y|x) = \frac{\exp(\sum_{i=1}^{m} \lambda_i f_i(x, y))}{\sum_y \exp(\sum_{i=1}^{m} \lambda_i f_i(x, y))}$$  (6)
Notion of Weights  Each $\lambda_i$ can be viewed as weight associated with every feature $f_i$. The training problem is to find these weights.
**Notion of Weights** Each $\lambda_i$ can be viewed as weight associated with every feature $f_i$. The training problem is to find these weights.

**Input** Feature Functions $f_1, f_2, ..., f_m$; empirical distribution $Pr^\sim(x, y)$

**Output** Optimal Parameters values $\Lambda^*$; optimal model $Pr_{\Lambda^*}(y|x)$

repeat

for $i = 1$ to $m$ do

$\delta_i$ be solution to

$$\delta_i = \frac{1}{T} \log \frac{\sum_{x,y} Pr^\sim(x, y)f_i(x,y)}{\sum_x Pr^\sim(x) \sum_y Pr_{\Lambda}(y|x)f_i(x,y)}$$

update values as $\lambda_i^{n+1} \leftarrow \lambda_i^n + \delta_i$

until $\Lambda$ is converged
• Viterbi Algorithm Used

alpha probability

\[ \alpha_{t+1}(y) = \sum_{y' \in Y} \Pr(y|\mathbf{x}_{t+1}) \alpha_{t}(y') \]

beta probability

\[ \beta_{t}(y') = \sum_{y \in Y} \Pr(y|\mathbf{x}_{t}) \beta_{t+1}(y) \]

delta probability

\[ \delta_{t+1}(y) = \max_{y' \in Y} \Pr(y|\mathbf{x}_{t+1}) \delta_{t}(y') \]

\(^aUsing same notation as McCallum et.al., 2000\)
### Application Specific Features

<table>
<thead>
<tr>
<th>Condition</th>
<th>Features</th>
</tr>
</thead>
<tbody>
<tr>
<td>$x_i$ is not rare</td>
<td>$x_i = W, y_i = K$</td>
</tr>
<tr>
<td>$x_i$ is rare</td>
<td>$W$ is prefix of $x_i$, $</td>
</tr>
<tr>
<td></td>
<td>$W$ is suffix of $x_i$, $</td>
</tr>
<tr>
<td></td>
<td>$x_i$ contains number, $y_i = K$</td>
</tr>
<tr>
<td></td>
<td>$x_i$ contains upper case character, $y_i = K$</td>
</tr>
<tr>
<td></td>
<td>$x_i$ contains hyphen, $y_i = K$</td>
</tr>
<tr>
<td></td>
<td>$x_{i-1}$ is “have”, “to”, “be”</td>
</tr>
</tbody>
</table>

| $\forall x_i$                 | $y_{i-1} = \bar{T}, y_i = K$                                            |
|                               | $y_{i-2}y_{i-1} = T_2 T_1, y_i = K$                                     |
|                               | $x_{i-2} = W, y_i = K$                                                  |
|                               | $x_{i-1} = W, y_i = K$                                                  |

**Table:** Table summarizing features used by [Ratnaparakhhi,96]
Graphical Models

1. Maximum Entropy Markov Model
2. Cyclic Dependency Network
3. Conditional Random Field
Label Bias Problem

- Particular label is biased towards some next state label

**Figure:** A. Transitions with un-normalized scores B. Transitions with probability values
Label Bias Problem

- Particular label is biased towards some next state label
- do not normalize scores per state

**Figure:** A. Transitions with un-normalized scores  B. Transitions with probability values
Conditional Random Field is defined as Markov Random Field globally conditioned on X
Conditional Random Field is defined as Markov Random Field globally conditioned on X

\[
\Pr(Y|X) = \frac{\Pr(X, Y)}{\Pr(X)} = \frac{\Pr(X, Y)}{\sum_{Y} \Pr(X, Y)} = \frac{\exp(\sum_{c \in C} V_c(X_c, Y_c))}{\sum_{Y} \exp(\sum_{c \in C} V_c(X_c, Y_c))} = \frac{\exp(\sum_{j=1}^{n} \sum_{i=1}^{M} \lambda_i f_i(y_{j-1}, y_j, X, j))}{Z(X)}
\]
Parameter Estimation

Voted Perceptron

\[ \Delta^t = F(Y, X) - F(Y^*, X) \]

where \( Y^* \) is best scoring labels obtained using model.

\[ \Lambda^{t+1} = \Lambda^t + \frac{1}{k} \sum_i \delta^t_i \]
**Inferencing**

1. **Initialization:** The values for all steps from the start state $\bot$ to all possible first states $s$ are set to the corresponding factor value.

$$\forall s \in S : \delta_1(s) = V_1(X, \bot, s)$$
$$= \psi_1(s) = \bot$$

2. **Recursion:** The values for the next steps are computed from the current value and the maximum values regarding all possible succeeding states $s'$.

$$\forall s \in S : 1 \leq j \leq n : \delta_j(s) = \max_{s' \in S} \delta_{j-1}(s') V_j(X, s', s)$$
$$\psi_j(s) = \arg \max_{s' \in S} \delta_{j-1}(s') V_j(X, s', s)$$
Termination:

\[ p^* = \max_{s' \in S} \delta_n(s') \]
\[ y_n^* = \arg \max_{s' \in S} \delta_n(s') \]

Path Backtracking: Recomputing the optimal path from the lattice using the track keeping values \( \psi_t \)

\[ y_t^* = \psi_{t+1}(y_{t+1}) \]

where \( t = n - 1, n - 2, \ldots, 1 \)