Maximum Entropy Markov Models for Information Extraction and Segmentation

Andrew McCallum, Dayne Freitag, Fernando Pereira, ICML 2000

Presented by
Ruogu Fang

Cornell University
Department of Electrical and Computer Engineering

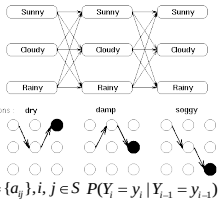
Outline

- Conditional Model
  - Hidden Markov Model
  - Maximum Entropy
- Maximum Entropy Markov Model Framework
  - Model :
    - Exponential Model for Transition
  - Learning:
    - Generalized Iterative Scaling
  - Prediction:
    - State estimation from Observations
- Experimental Results

Hidden Markov Model (HMM)

- A hidden Markov model (HMM) is a triple (Π, A, B)
- States: $y \in \{s_1, \ldots, s_k\}$
- Output symbols: $x \in \{o_1, \ldots, o_m\}$
- Initial State Probabilities $\Pi = \{\pi_i\}, i \in S \quad P(Y_1 = y_i)$
  - Specifies where the sequence starts
- State Transition Probabilities $A = \{a_{ij}\}, i, j \in S \; P(Y_i = y_i \mid Y_{i-1} = y_{i-1})$
  - Probability that one state succeeds another
- Symbol Emission Probabilities $B = \{b_{ij}\}, i, j \quad P(X_i = x_i \mid Y_i = y_i)$
  - Probability that observation is generated in this state
- Every output + state sequence has a probability

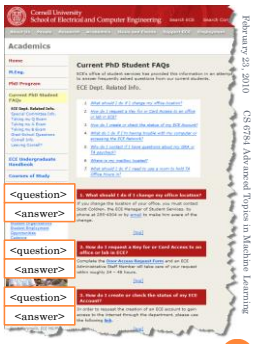
$$P(x_1, \ldots, x_n, y_1, \ldots, y_n) = [P(y_1)P(x_1 \mid y_1)\prod_{i=2}^{I} P(x_i \mid y_i)P(x_i \mid y_{i-1})]$$

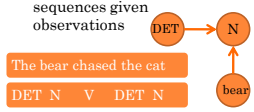


Limitation ?

Objective of MEMM

- Maximum Entropy: Incorporate multiple overlapping features
  - Indentation
  - Numbered questions
  - Styles of paragraph breaks
  - Line length
- HMM: Conditional probability of state sequences given observations





Experiment Setup

Experimental Data

- 38 les belonging to 7 Usenet multipart FAQs
- <head>X-NNTP-Poster: NewsHound v1.33
- <head>
- <head>Archive-name: acorn/faq/part2
- <head>Frequency: monthly
- <head>
- <question>2.6) What configuration of serial cable should I use
- <answer>
- <answer> Here follows a diagram of the necessary connections
- <answer>programs to work properly. They are as far as I know t
- <answer>agreed upon by commercial comms software developers fo
- <answer>
- <answer> Pins 1, 4, and 8 must be connected together inside
- <answer>is to avoid the well known serial port chip bugs. The
- *Procedure: For each FAQ, train on one part, test on others; average*

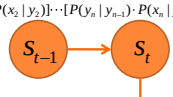
Features in Experiment

- begins-with-number
- begins-with-ordinal
- begins-with-punctuation
- begins-with-question-word
- begins-with-subject
- blank indented-1-to-4
- contains-alphanum
- contains-bracketed-number
- contains-http
- contains-non-space
- contains-number
- contains-pipe
- contains-question-mark
- contains-question-word
- ends-with-question-mark
- rst-alpha-is-capitalized
- Indented
- indented-1-to-4
- indented-5-to-10
- more-than-one-third-space
- only-punctuation
- prev-is-blank
- prev-begins-with-ordinal
- shorter-than-30

Maximum Entropy Markov Models

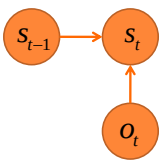
HMM

$$P(y,x) = [P(y_1) \cdot P(x_1 \mid y_1)][P(y_2 \mid y_1) \cdot P(x_2 \mid y_2)] \cdots [P(y_n \mid y_{n-1}) \cdot P(x_n \mid y_n)]$$



MEMM

$$P(y \mid x) = [P(y_1 \mid x_1, y_0)][P(y_2 \mid x_2, y_1)] \cdots [P(y_n \mid x_n, y_{n-1})]$$



Transition Prob $P(s_t \mid s_{t-1})$
Emission Prob $P(o_t \mid s_t)$

$P(s_t \mid s_{t-1}, o_t)$

$P_{s_{t-1}}(s_t \mid o)$

$|S|$

Exponential Model for Transition

- Model transition in terms of multiple, non-independent features of observations
- Objective function: H(p)
- Goal: Among all the distributions that satisfy the constraints, choose the one, p*, that maximizes H(p).

$$p^* = \arg\max_{p \in P} H(p)$$

- Question: How to represent constraints?
- Feature: a binary valued function on events

$$f_j : \varepsilon \to \{0,1\}, \quad \varepsilon = A \times B$$

- A: the set of possible classes (e.g., tags in POS tagging)
- B: space of contexts (e.g., neighboring words/ tags in POS tagging)

$$f_j(a,b) = \begin{cases} 1 & \text{if } a = DET \ \& \ curWord(b) = \text{"that"} \\ 0 & \text{o.w.} \end{cases}$$

In courtesy of Fei Xia, Maximum Entropy Model, 02/2006

Exponential Model for Transition (cont)

- Task: Find p*

$$p^* = \arg\max_{p \in P} H(p)$$

where $P = \{p \mid E_p f_j = E_{\tilde{p}} f_j, j = \{1,...,k\}\}$

- Objective function: H(p)
- Constraints: $\{E_p f_j = E_{\tilde{p}} f_j = d_j, j = \{1,...,k\}\}$

$\sum_x p(x) = 1$ ⇒ Add a feature $f_0(a,b) = 1 \quad \forall a,b$

- Is P empty? $E_p f_0 = E_{\tilde{p}} f_0 = 1$
- Questions:
  - Does p* exist?
  - Is p* unique?
  - What is the form of p*?
  - How to find p*?

$$p(x) = \frac{e^{\sum_{j=1}^{k} \lambda_j f_j(x)}}{Z}$$

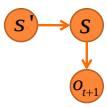
Z is the normalizing factor that makes the distribution sum to one across all next states s

State Estimation

HMM

- forward probability
  - probability of producing the observation sequence up to time t and being in state s at time t

$$\alpha_{t+1}(s) = \sum_{s' \in S} \alpha_t(s') \cdot P(s \mid s') \cdot P(o_{t+1} \mid s)$$



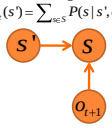
MEMM

- Forward probability $\alpha_t(s)$
  - Probability of being in state s at time t given the observation sequence up to time t

$$\alpha_{t+1}(s) = \sum_{s' \in S} \alpha_t(s') \cdot P_{s'}(s \mid o_{t+1})$$

- Backward probability

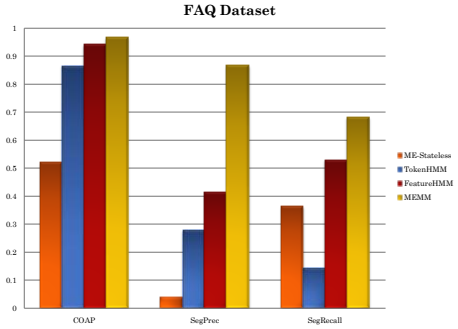
$$\beta_t(s') = \sum_{s \in S} P(s \mid s', o_t) \cdot \beta_{t+1}(s)$$



Models Tested

- **MaxEntStateless**
  - A single maximum entropy classier applied to each line independently.
- ***TokenHMM***
  - A fully connected HMM with four states, one for each of the line categories, each of which generates individual tokens (groups of alphanumeric characters and individual punctuation characters).
- ***FeatureHMM***
  - Identical to *tokenHMM*, only the lines in a document are first converted to sequences of features.
- ***MaxEntMM***
  - The maximum entropy Markov model described in this paper.

Experimental Results



COAP $P_\mu(act, pred) = \sum_{i,j} D_\mu(i,j) \delta_{act}(i,j) \oplus \delta_{pred}(i,j)$

Summary

- Combine HMMs and maximum-entropy models into a general model
  - Allow state transitions to depend on non-independent features of the sequence under analysis
  - Conditional model that represents the probability of reaching a state given an observation and the previous state
- Training algorithm
  - Direct generalizations of HMMs algorithms: forward-backward, Baum-Welch
  - Generalized iterative scaling for $P(s \mid s', o)$
  - Generalized Iterative Scaling (GIS)
- Prediction: Dynamic programming

Additional Slides

Maximum Entropy

- Goal: Estimate p
- Choose p with maximum entropy (or "uncertainty") subject to the constraints (or "evidence")

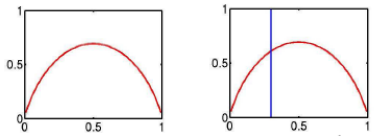
$$H(p) = -\sum_{x \in A \times B} p(x) \log p(x) \quad x = (a,b), \quad where \quad a \in A \wedge b \in B$$

- Training
  - a: state to be predicted
  - b: context
  - Example: a=NN, b=tags for current and previous words
  - Learn probabilities of each (a,b): P(a,b)
- Maximum Entropy = Minimum Commitment
  - Model all is known: satisfy a set of constraints
  - Assume nothing about what is unknown: choose the most "uniform" distribution

Maximum Entropy: Coin Flip

- Toss a coin: P(H) = p1, P(T) = p2
- Constraint: p1 + p2 = 1
- Question: What is your estimation of p=(p1,p2)?
- Answer: Choose the p that maximizes H(p)

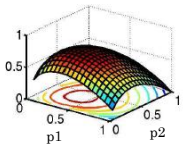
$$H(p) = -\sum_{x} p(x) \log p(x)$$



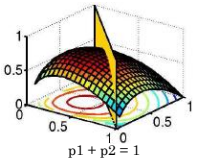
p1 p1=0.3

In courtesy of Fei Xia, Maximum Entropy Model, 02/2006

Maximum Entropy: Coin Flip

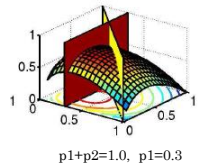


p1 p2



p1 + p2 = 1

- Maximum Entropy = Minimum Commitment
  - Model all is known: satisfy a set of constraints
  - Assume nothing about what is unknown: choose the most "uniform" distribution



p1+p2=1.0, p1=0.3

In courtesy of Fei Xia, Maximum Entropy Model, 02/2006

Maximum Entropy Markov Model

Outline

- Conditional Model
  - Hidden Markov Model
  - Maximum Entropy
- Maximum Entropy Markov Model Framework
  - **Model :**
    - **Exponential Model for Transition**
  - Learning:
    - Generalized Iterative Scaling
  - Prediction:
    - State estimation from Observations
- Variations
- Experimental Results

Exponential Model for Transition (cont)

- Finite training sample of events: S
- Observed probability of x in S: $\tilde{p}(x)$
- The model p's probability of x: $p(x)$
- The j[th] feature: f_j
- Observed expectation of f_j $E_{\tilde{p}} f_j = \sum_{x \in \mathcal{E}} \tilde{p}(x) f_j(x)$
  - Empirical count of f_j
- Model expectation of f_j $E_p f_j = \sum_{x \in \mathcal{E}} p(x) f_j(x)$
- Model's feature expectation = observed feature expectation

$$E_p f_j = E_{\tilde{p}} f_j$$

- How to calculate $E_{\tilde{p}} f_j$? $E_{\tilde{p}} f_j = \sum_{x \in \mathcal{E}} \tilde{p}(x) f_j(x) = \frac{\sum_{i=1}^{N} f_j(x)}{N}$

$$f_j(a,b) = \begin{cases} 1 & \text{if } a = DET \ \& \ curWord(b) = \text{"that"} \\ 0 & \text{o.w.} \end{cases}$$

Outline

- Conditional Model
  - Hidden Markov Model
  - Maximum Entropy
- Maximum Entropy Markov Model Framework
  - Model :
    - Exponential Model for Transition
  - **Learning:**
    - **Generalized Iterative Scaling**
  - Prediction:
    - State estimation from Observations
- Variations
- Experimental Results

Parameter Estimation

Generalized Iterative Scaling (Darroch & Ratcliff, 1972)

- Setup
  - Obey form of model and constraints

$$p(x) = \frac{e^{\sum_{j=1}^{k} \lambda_j f_j(x)}}{Z} \qquad F_a = \frac{1}{m_s} \sum_{k=1}^{m_s} f_a(o_{t_k}, s_{t_k})$$

  - Additional constraint: $\forall x \in \mathcal{E} \quad \sum_{j=1}^{k} f_j(x) = C$
  - Let $C = \max_{x \in \mathcal{E}} \sum_{j=1}^{k} f_j(x)$
  - Add a new feature f_{k+1}: $\forall x \in \mathcal{E} \quad f_{k+1}(x) = C - \sum_{j=1}^{k} f_j(x)$

Generalized Iterative Scaling Algorithm

- Compute F_a, j=1, …, k+1
- Initialize $\lambda_j^{(0)} = 1$ (any values, e.g., 1)
- REPEAT until converge
  - FOR each j
    - Compute $E_{p^{(n)}} f_j = \sum_{x \in \mathcal{E}} p^{(n)}(x) f_j(x)$ **E-Step**

      where $p^{(n)}(x) = \frac{e^{\sum_{j=1}^{k+1} \lambda_j^{(n)} f_j(x)}}{Z}$

    - Update $\lambda_j^{(n+1)} = \lambda_j^{(n)} + \frac{1}{C}(\log \frac{F_a}{E_{p^{(n)}} f_j})$ **M-Step**
  - END FOR
- END REPEAT

What is the Form of p*

(Ratnaparkhi, 1997)

$$P = \{p \mid E_p f_j = E_{\tilde{p}} f_j, j = \{1,...,k\}\}$$

$$Q = \{p \mid p(x) = \pi \prod_{j=1}^{k} \alpha_j^{f_j(x)}, \alpha_j > 0\}$$

- Theorem: if $p^* \in P \cap Q$ then $p^* = \arg\max_{p \in P} H(p)$
- Furthermore, p* is unique.

Approximation for Calculating Feature Expectation

$$E_p f_j = \sum_{x \in \mathcal{E}} p(x) f_j(x) = \sum_{a \in A, b \in B} p(a,b) f_j(a,b)$$

$$= \sum_{a \in A, b \in B} p(b) p(a \mid b) f_j(a,b)$$

$$\approx \sum_{a \in A, b \in B} \tilde{p}(b) p(a \mid b) f_j(a,b)$$

$$= \sum_{b \in B} \tilde{p}(b) \sum_{a \in A} p(a \mid b) f_j(a,b)$$

$$= \frac{1}{N} \sum_{i=1}^{N} \sum_{a \in A} p(a \mid b_i) f_j(a, b_i)$$

Properties of GIS

- $L(p^{(n+1)}) >= L(p^{(n)})$
- The sequence is guaranteed to converge to p*.
- The converge can be very slow.

- The running time of each iteration is O(NPA):
  - N: the training set size
  - P: the number of classes
  - A: the average number of features that are active for a given event (a, b).

Outline

- Conditional Model
  - Hidden Markov Model
  - Maximum Entropy
- Maximum Entropy Markov Model Framework
  - Model :
    - Exponential Model for Transition
  - Learning:
    - Generalized Iterative Scaling
  - **Prediction:**
    - **State estimation from Observations**
- Variations
- Experimental Results

Experimental Results