

An Alternate Objective Function for Markovian Fields

Paper by Sham Kakade, Yee Whye Teh,
and Sam T. Roweis, ICML 2002

Presentation by Yohan Kim

May 28, 2002

Introduction

- Input-output modeling of sequential data is a fundamental problem.
- Task is to infer 'state' or 'label' sequence $S = \{s_1, s_2, \dots, s_T\}$ given some observations $X = \{x_1, x_2, \dots, x_T\}$.
- Learning approach is to model certain aspects of the joint distribution over states and observations.
- New architectures have been proposed. They model only the conditional distribution of state sequences given the features.
- Features are certain properties of the state and observation sequence.
- Little work exists on the choice of objective functions in these models.

Outline

- Old objective function
- The new objective function
- Definitions of MEMM and CRF
- Old and new MEMM
- Old and new CRF
- Experiment: blind robot example
- Parameter estimation for MEMM and CRF

Old objective function

We maximize the log probability of the *entire* state sequence given the observation sequence:

$$C_0(\theta; S, X) = \log p(S | X; \theta) = \log p(s_T, s_{T-1}, \dots, s_1 | X; \theta)$$

Lemma: Assuming a Markov property,

$$C_0(\theta; S, X) = \sum_t \log p(s_t | s_{t-1}, X; \theta)$$

Proof:

Using the property $p(A, B) = p(A|B)p(B)$

$$p(s_T, s_{T-1}, \dots, s_1 | X; \theta) = p(s_T | s_{T-1}, \dots, s_1, X; \theta) p(s_{T-1} | s_{T-2}, \dots, s_1, X; \theta) \dots p(s_2 | s_1, X; \theta)$$

Using the property of Markov chain

$$p(s_T, s_{T-1}, \dots, s_1 | X; \theta) = p(s_T | s_{T-1}, X; \theta) p(s_{T-1} | s_{T-2}, X; \theta) \dots p(s_2 | s_1, X; \theta)$$

End of proof.

The new objective function

Alternatively we can maximize average single time prediction cost:

$$C_1(\theta; S, X) = \frac{1}{T} \sum_t \log p(s_t | X; \theta)$$

This objective function emphasizes making the most probable choice at each time even though the joint probability may be low.

Definitions of MEMM and CRF

Maximum Entropy Markov Model (MEMM)

$$p(S | X, \theta) = \prod_{t=1}^T p(s_t | s_{t-1}, x_t)$$

$$p(s_t | s_{t-1}, x_t) = \frac{1}{Z(s_{t-1}, x_t)} \exp\left(\sum_k \lambda_k f_k(s_t, s_{t-1}, x_t)\right)$$

$$\theta = \{\lambda_1, \lambda_2, \dots, \lambda_k\}$$

Z = normalizing factor

x_t = observation at time t ; it can also be a feature of the entire observation sequence

Conditional Random Field (CRF)

$$p(S | X, \theta) = \frac{1}{Z(S, X)} \prod_{t=1}^T \exp\left(\sum_k \lambda_k f_k(s_t, s_{t-1}, x_t)\right)$$

The crucial difference is that CRF has a global normalizing $Z(S, X)$ factor whereas MEMM has a local one, $Z(s_{t-1}, x_t)$.

Global normalizing factor gives CRF a joint distribution and makes it undirected model.

Definitions of MEMM and CRF, cont.



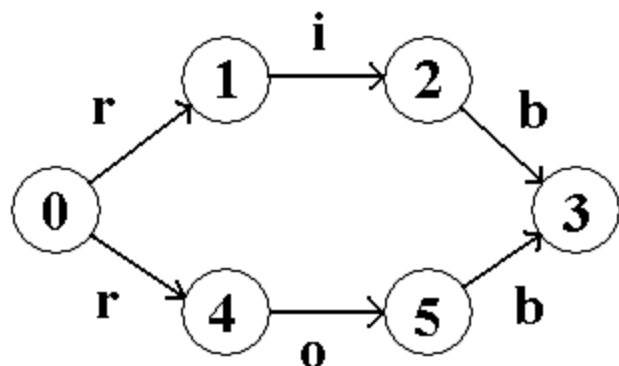
Under MEMM, probability mass is assigned to s_t by looking at a particular transition function $p(s_t | s_{t-1}, x_t)$ that takes as input a fixed previous state s_{t-1} and an observation.

CRF uses only one joint function, $p(S|X)$, to infer states. This is accomplished by looking at neighboring states and observations *past/present* before inferring anything about the current state.

Old and new MEMM

Under $C_0(\theta; S, X) = \sum_t \log p(s_t | s_{t-1}, \dots, s_1, X; \theta)$

MEMM cannot accurately estimate previously unobserved transitions since these are not included in the old objective function. This is illustrated in the ‘label-bias’ problem.



Training sequence consists of observation sequence ‘r i b’ labeled ‘0123’ and ‘r o b’ labeled ‘0453’.

During training, $p(2|1,i)$ is used but not $p(2|1,o)$ since the latter is not observed in the training data. This results in incorrect estimation of $p(2|1,o)$.

Old and new MEMM, cont.

Under the new objective function,

$$C_1(\theta; S, X) = \frac{1}{T} \sum_t \log p(s_t | X; \theta)$$

Using the law of total probability,

$$p(s_t | x_1^t) = \sum_{s_{t-1}} p(s_t | s_{t-1}, x_t) p(s_{t-1} | x_1^{t-1})$$

Old and new CRF

Difference under two objective functions is subtle since CRF doesn't suffer from the 'label-bias' problem.

Consider another didactic example:

label : 12123434321212124343434321 24343412121
observation: aaaabbbbbbaaaaaabbbbbbbbaaabbbbbbaaaaa

We spend 50% of time in each of two modes (say, 1st and 2nd) and switch between them with equal probability. While in

- 1st mode, label alternates between 1 and 2 and observation is 'a'.
- 2nd mode, label alternates between 3 and 4 and observation is 'b'.

Old and new CRF, cont.

We construct an impoverished model with only one feature function $f(s_i, s_{i-1}, x_i)$ with following properties:

$$f(s_i, s_{i-1}, x_i) = \begin{cases} 1 & \text{if } (1,1, a), (2,2, a), (3,3, b), \text{ or } (4,4, b) \\ 0 & \text{if otherwise} \end{cases}$$

When training CRF with this feature, a large weight λ is assigned to this feature function.

The reason is that the empirical expectation of this feature (zero in this case) should equal to that calculated using the trained model.

Old and New CRF, cont.

During testing, error rate is approximately 100%. To understand this, consider when the observation sequence is 'aa'.

If the trained model is to label this sequence, it has 8 choices:

'11', '22', '12', '21', '33', '44', '34', '43'

Because of the large negative weight, '11' and '22' are assigned very low probabilities. The others are assigned equal probability.

possible labels

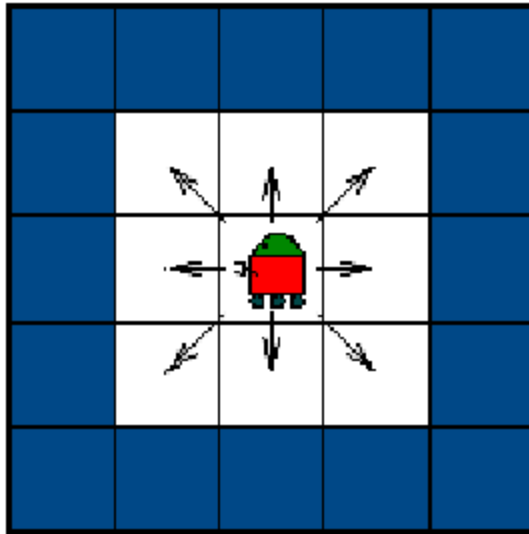
12 **21** **33** **44** **34** **43**

Labels '11' and '22' are very unlikely.

Size of the box \propto probability mass

Under C_1 , large positive weight is assigned because 'this gives a perfect conditional distribution.' Error rate \approx 50%.

Experiment: Blind Robot Example



The robot can take any one of 8 directions. When the robot is in a blue box, it detects a wall.

A robot is in a grid world.

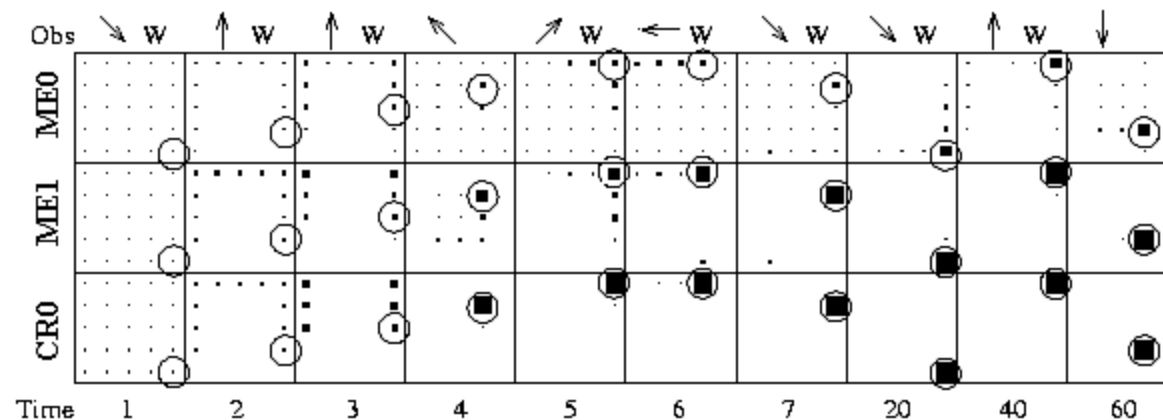
Task is to predict the location of the robot given a sequence of sensor readings, a sequence of movements and a uniform distribution of starting location.

This experiment compares three models:

1. ME0 = MEMM with C_0
2. ME1 = MEMM with C_1
3. CR0 = CRF with C_0

(with C_1 , results are similar)

Experiment: Blind Robot Example, cont.

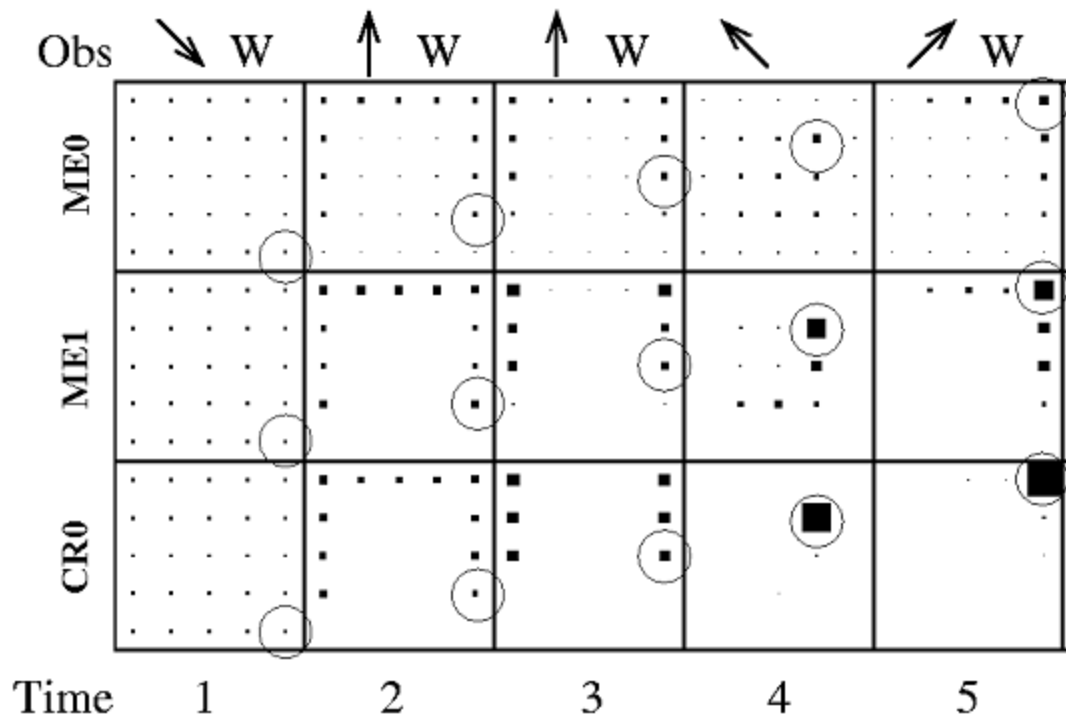


Observation sequence: $X = \{x_1, x_2, \dots, x_T\}$, where $x_t = \{\text{last direction taken, whether a wall is present}\}$ at time t .

For each time step, each model has to assign probabilities to positions on the grid. That is, it only sees past and present but not future.

Correct positions are indicated by the circles. Size of the blobs placed on the grid correspond to probability mass.

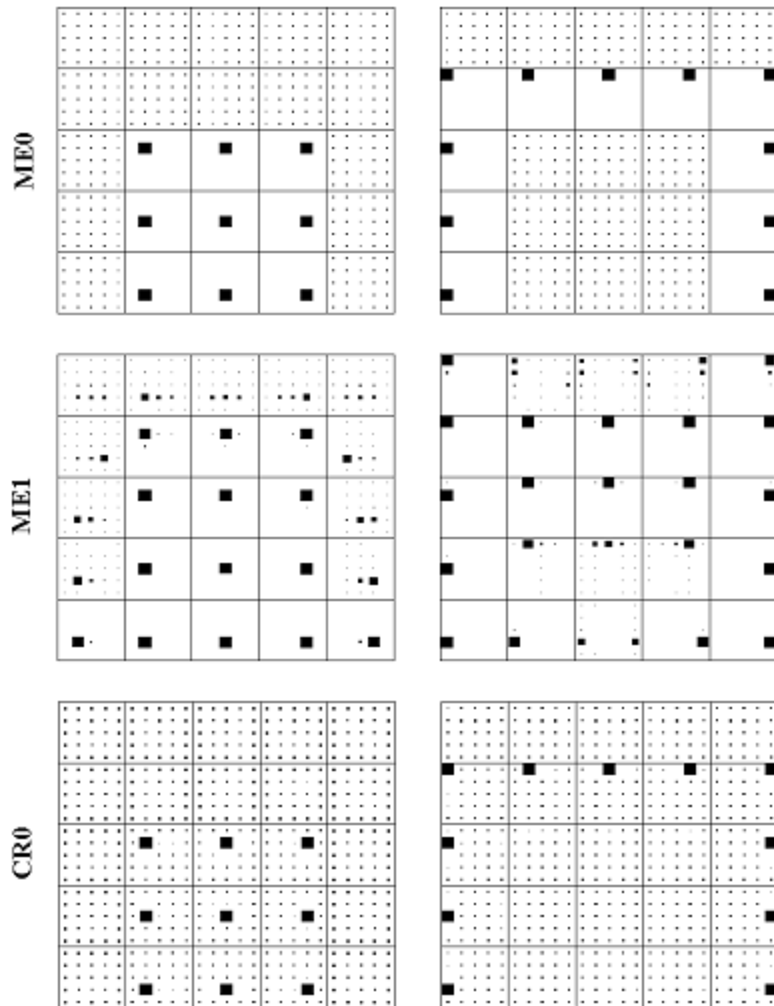
Experiment: Blind Robot Example, cont.



As indicated by the size of the blobs, both ME1 and CR0 perform quite well. Compared to ME0, they assign much greater probability mass to the correct positions of the robot.

Experiment: Blind Robot Example, cont.

move north, no wall move north, finds wall



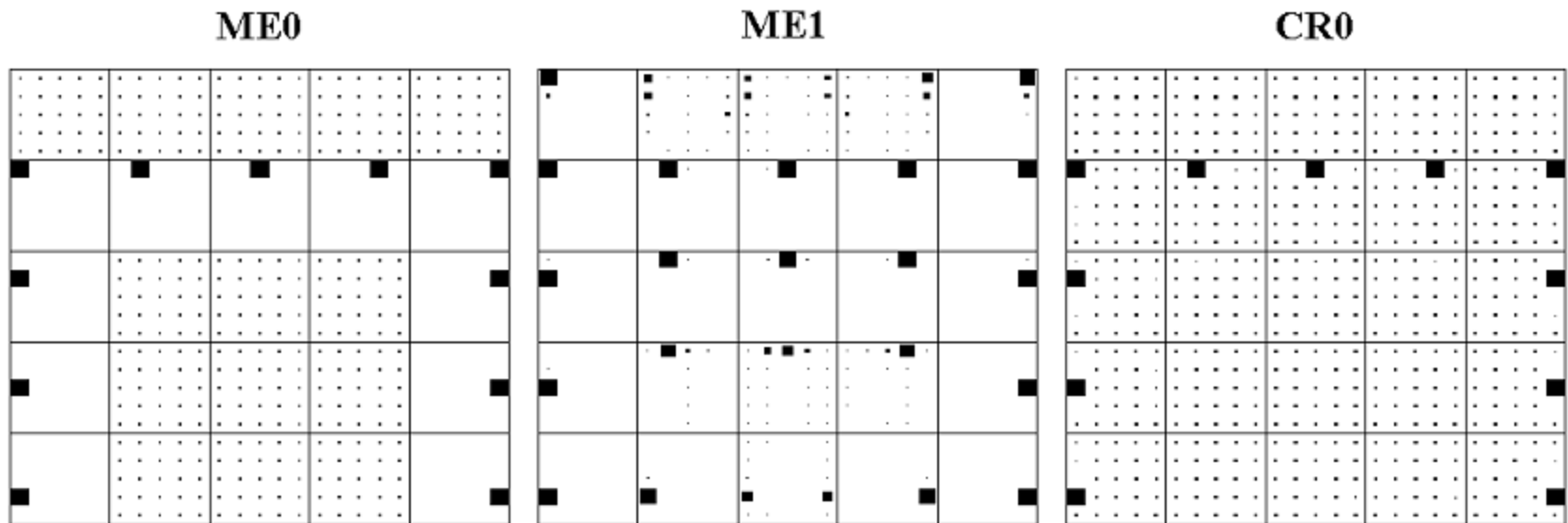
Features learned by the three models.

Position of each square on the grid corresponds to the previous location.

Position of each blob within each square is the next location.

Experiment: Blind Robot Example, cont.

Features learned by the three models for {move north, wall} case.



When previous current locations are consistent with the previous location, ME0 and CR0 learned to move correctly. Otherwise no learning occurred. ME1 assigns a likely location even when previous location is inconsistent.

Parameter Estimation for MEMM and CRF

Under objective function C_0 , training is carried out by maximizing $\log P(S|X)$, which results in equating empirical expectations of features with those calculated using the model.

The derivative of C_1 with respect to parameter of interest is calculated and the new parameter can be updated using conjugate gradient.

Parameter Estimation for MEMM and CRF, cont.

The derivative of C_1 is as follows:

$$\frac{\partial C_1}{\partial \lambda_k} = \frac{1}{T} \sum_{t,j} \langle f_k(s_t, s_{t-1}, x_t) \rangle_{p(s_t, s_{t-1} | s_t = l_t, X, \theta)} - \sum_i \langle f_k(s_i, s_{i-1}, x_i) \rangle_{p(s_i, s_{i-1} | X, \theta)}$$

Brackets mean taking expectation of the argument with the subscripted probability function.

We divide the task of calculating the above quantity into two parts corresponding to each term.

Let $w_i(l_i, l_{i-1}) = p(s_i = l_i, s_{i-1} = l_{i-1} | X, \theta)$

Second term is calculated by first getting $w_i(l_i, l_{i-1})$ using belief propagation and then calculating the expectation of the features explicitly.

Parameter Estimation for MEMM and CRF, cont.

$$\frac{\partial C_1}{\partial \lambda_k} = \frac{1}{T} \sum_{tj} \langle f_k(s_i, s_{i-1}, x_i) \rangle p(s_i, s_{i-1} | s_t = l_t, X, \theta) - \sum_i \langle f_k(s_i, s_{i-1}, x_i) \rangle p(s_i, s_{i-1} | X, \theta)$$

First term is then calculated by defining the term $\omega_i(l_i, l_{i-1})$ and the expectations of features are calculated similarly.

$$\omega_i(l_i, l_{i-1}) = \sum_{t=1}^T p(s_i = l_i, s_{i-1} = l_{i-1} | s_t = \hat{l}_t, X, \theta)$$

$$\omega_i(l_i, l_{i-1}) = \omega_i^f(l_i, l_{i-1}) + \omega_i^b(l_i, l_{i-1})$$

$$\omega_i^f(l_i, l_{i-1}) = \sum_{t=1}^{i-1} p(s_i = l_i, s_{i-1} = l_{i-1} | s_t = \hat{l}_t, X, \theta)$$

$$\omega_i^b(l_i, l_{i-1}) = \sum_{t=i}^T p(s_i = l_i, s_{i-1} = l_{i-1} | s_t = \hat{l}_t, X, \theta)$$

Parameter Estimation for MEMM and CRF, cont.

$$\omega_{i+1}^f(l_{i+1}, l_i) = w_{i+1}(s_{i+1} = l_{i+1} \mid s_i = l_i) v_{i+1}^f(l_i)$$

$$v_{i+1}^f(l_i) = \begin{cases} \delta_{l_i, \hat{l}_i} & i = 1 \\ \delta_{l_i, \hat{l}_i} + \sum_{l_{i-1}} \omega_i^f(l_i, l_{i-1}) & i > 1 \end{cases}$$

$$\omega_i^b(l_i, l_{i-1}) = w_i(s_i = l_i, s_{i-1} = l_{i-1}) v_i^b(l_i)$$

$$v_i^b(l_i) = \begin{cases} \delta_{l_i, \hat{l}_i} & i = T \\ \delta_{l_i, \hat{l}_i} + \sum_{l_{i+1}} \omega_{i+1}^b(l_{i+1}, l_i) & i < T \end{cases}$$

Conclusions

The new objective function improves performances.

‘label-bias’ problem can be addressed by using the new objective function.