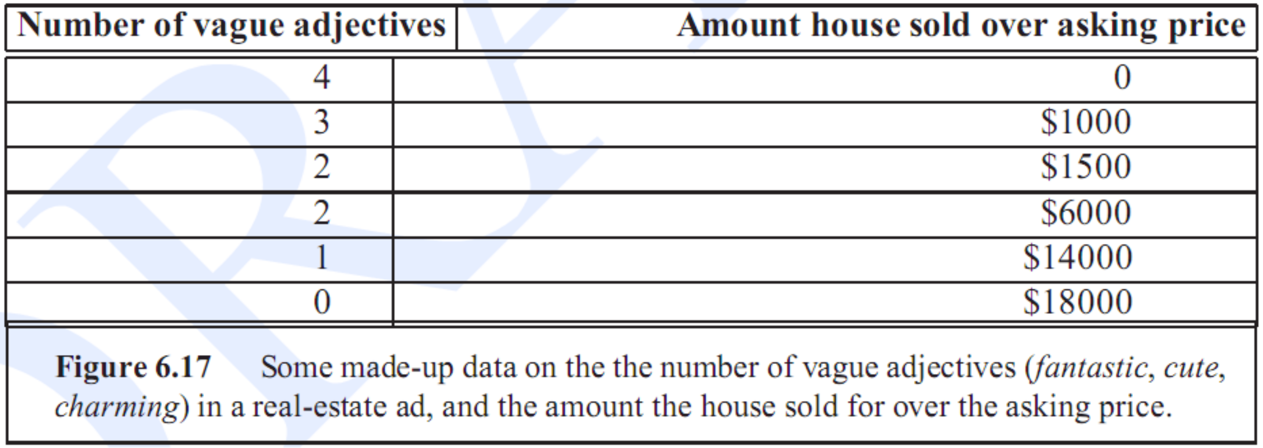
Maximum Entropy Model

Introduction

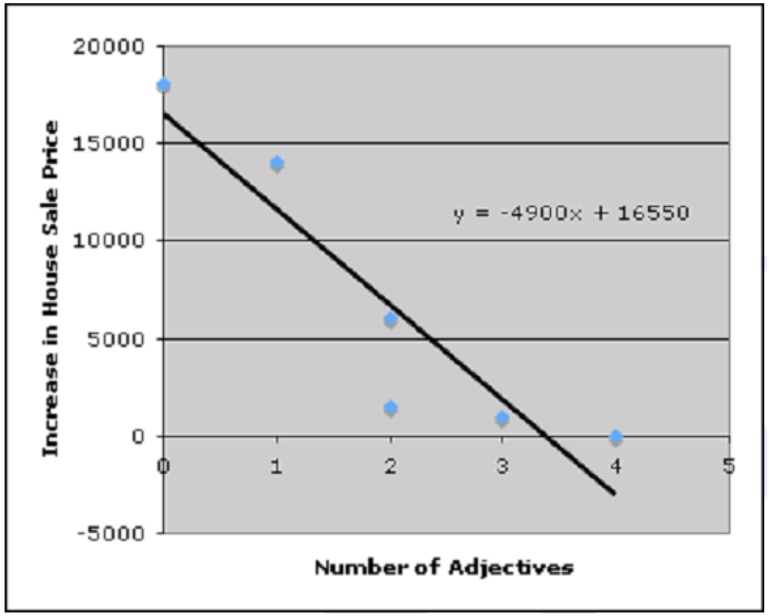
- Maximum Entropy (MaxEnt)
  - More widely known as multinomial logistic regression
- Begin with non-sequential classifier
  - A probabilistic classifier
  - Exponential or log-linear classifier
  - Text classification
  - Sentiment analysis
    - Positive or negative opinion
  - Sentence boundary

Linear Regression



Number of vague adjectives	Amount house sold over asking price
4	0
3	\$1000
2	\$1500
2	\$6000
1	\$14000
0	\$18000

Figure 6.17 Some made-up data on the the number of vague adjectives (*fantastic*, *cute*, *charming*) in a real-estate ad, and the amount the house sold for over the asking price.



Linear Regression

$$\text{price} = w_0 + w_1 * \text{Num\_Adjectives} + w_2 * \text{Mortgage Rate} + w_3 * \text{Num\_Unsold\_Houses}$$

$$\text{price} = w_0 + \sum_{=1}^{N} w_i \times f_i$$

$$y = \sum_{i=0}^{N} w_i \times f_i$$

$$y = w \cdot f$$

$$y_{pred}^{(j)} = \sum_{i=0}^{N} w_i \times f_i^{(j)}$$

$$\text{cost}(W) = \sum_{j=0}^{M} \left(y_{pred}^{(j)} - y_{obs}^{(j)} \right)^2$$

sum square error

- $x^{(j)}$: a particular instance
- $y^{(j)}_{obs}$: observed label in the training set of $x^{(j)}$
- $y^{(j)}_{pred}$: predicted value from linear regression model

Logistic Regression – simplest case of binary classification

- Consider whether x is in class (1, true) or not (0, false)

$$\begin{aligned}\underset{\in [0,1]}{\boxed{P(y=true|x)}} &= \sum_{i=0}^{N} w_i \times f_i \\ &= w \cdot f \qquad w \cdot f \in (-\infty,\infty)\end{aligned}$$

$$\underset{\in [0,\infty)}{\boxed{\frac{p(y=true)|x}{1-p(y=true|x)}}} = w \cdot f$$

$$\underset{\in (-\infty,\infty)}{\boxed{\ln\left(\frac{p(y=true|x)}{1-p(y=true|x)}\right)}} = w \cdot f \qquad \operatorname{logit}(p(x)) = \ln\left(\frac{p(x)}{1-p(x)}\right)$$

Logistic Regression – simplest case of binary classification

$$\ln\left(\frac{p(y=\text{true}|x)}{1-p(y=\text{true}|x)}\right)=w\cdot f$$

$$\frac{p(y=\text{true}|x)}{1-p(y=\text{true}|x)}=e^{w\cdot f}$$

$$p(y=\text{true}|x)=(1-p(y=\text{true}|x))e^{w\cdot f}$$

$$p(y=\text{true}|x)=e^{w\cdot f}-p(y=\text{true}|x)e^{w\cdot f}$$

$$p(y=\text{true}|x)+p(y=\text{true}|x)e^{w\cdot f}=e^{w\cdot f}$$

$$p(y=\text{true}|x)(1+e^{w\cdot f})=e^{w\cdot f}$$

$$\boxed{\begin{aligned}p(y=\text{true}|x)&=\frac{e^{w\cdot f}}{1+e^{w\cdot f}}\\ p(y=false|x)&=\frac{1}{1+e^{w\cdot f}}\end{aligned}}$$

$$\boxed{\begin{aligned}p(y=\text{true}|x)&=\frac{\exp(\Sigma_{i=0}^{N}w_if_i)}{1+\exp(\Sigma_{i=0}^{N}w_if_i)}\\ p(y=\text{false}|x)&=\frac{1}{1+\exp(\Sigma_{i=0}^{N}w_if_i)}\end{aligned}}$$

$$\boxed{\begin{aligned}p(y=\text{true}|x)&=\frac{e^{w\cdot f}}{1+e^{w\cdot f}}\\ &=\frac{1}{1+e^{-w\cdot f}}\\ p(y=\text{false}|x)&=\frac{e^{-w\cdot f}}{1+e^{-w\cdot f}}\end{aligned}}$$

Logistic Regression – Classification

$$p(y = true|x) > p(y = false|x)$$

$$\frac{p(y = true|x)}{p(y = false|x)} > 1$$

$$\frac{p(y = true|x)}{1 - p(y = true|x)} > 1$$

$$e^{w \cdot f} > 1$$

$$w \cdot f > 0$$

$$\sum_{i=0}^{N} w_i f_i > 0$$

Advanced: Learning in logistic regression

Conditional Maximum Likelihood Estimation

$$\hat{w} = \underset{w}{\operatorname{argmax}}\, P(y^{(i)}|x^{(i)})$$

$$\hat{w} = \underset{w}{\operatorname{argmax}} \prod_i P(y^{(i)}|x^{(i)})$$

$$\hat{w} = \underset{w}{\operatorname{argmax}} \sum_i \log P(y^{(i)}|x^{(i)})$$

$$\hat{w} = \underset{w}{\operatorname{argmax}} \sum_i \log \begin{cases} P(y^{(i)}=1|x^{(i)})) & \text{for } y^{(i)}=1 \\ P(y^{(i)}=0|x^{(i)})) & \text{for } y^{(i)}=0 \end{cases}$$

$$\hat{w} = \underset{w}{\operatorname{argmax}} \sum_i y^{(i)} \boxed{\log P(y^{(i)}=1|x^{(i)})} + (1-y^{(i)}) \boxed{\log P(y^{(i)}=0|x^{(i)})}$$

$$\hat{w} = \underset{w}{\operatorname{argmax}} \sum_i y^{(i)} \boxed{\log \frac{e^{-w\cdot f}}{1+e^{-w\cdot f}}} + (1-y^{(i)}) \boxed{\log \frac{1}{1+e^{-w\cdot f}}}$$

$$\begin{aligned} p(y=\text{true}|x) &= \frac{e^{w\cdot f}}{1+e^{w\cdot f}} \\ &= \frac{1}{1+e^{-w\cdot f}} \\ p(y=\text{false}|x) &= \frac{e^{-w\cdot f}}{1+e^{-w\cdot f}} \end{aligned}$$

Maximum Entropy Modeling

- Input: x (a word need to tag or a doc need to classify)
  - Features
    - Ends in –ing
    - Previous word is "the"
  - Each feature f_i, weight w_i
  - Particular class *c*
  - *Z* is a normalizing factor, used to make the prob. sum to 1

$$p(c|x) = \frac{1}{Z}\exp(\sum_i w_i f_i)$$

Maximum Entropy Modeling

$$p(c|x) = \frac{1}{Z} \exp \sum_i w_i f_i$$

C = {c_1, c_2, …, c_C}

Normalization

$$Z = \sum_C p(c|x) = \sum_{c' \in C} \exp \left(\sum_{i=0}^{N} w_{c'i} f_i \right)$$

$$p(c|x) = \frac{\exp \left(\sum_{i=0}^{N} w_{ci} f_i \right)}{\sum_{c' \in C} \exp \left(\sum_{i=0}^{N} w_{c'i} f_i \right)}$$

$$f_1(c,x) = \begin{cases} 1 & \text{if } word_i = \text{“race”} \;\;\&\;\; c = \text{NN} \\ 0 & \text{otherwise} \end{cases}$$

$$f_2(c,x) = \begin{cases} 1 & \text{if } t_{i-1} = \text{TO} \;\;\&\;\; c = \text{VB} \\ 0 & \text{otherwise} \end{cases}$$

$$f_3(c,x) = \begin{cases} 1 & \text{if suffix}(word_i) = \text{“ing”} \;\;\&\;\; c = \text{VBG} \\ 0 & \text{otherwise} \end{cases}$$

$$f_4(c,x) = \begin{cases} 1 & \text{if is\_lower\_case}(word_i) \;\;\&\;\; c = \text{VB} \\ 0 & \text{otherwise} \end{cases}$$

$$f_5(c,x) = \begin{cases} 1 & \text{if } word_i = \text{”race”} \;\;\&\;\; c = \text{VB} \\ 0 & \text{otherwise} \end{cases}$$

$$f_6(c,x) = \begin{cases} 1 & \text{if } t_{i-1} = \text{TO} \;\;\&\;\; c = \text{NN} \\ 0 & \text{otherwise} \end{cases}$$

f_i: A feature that only takes on the values 0 and 1 is also called an indicator function

In MaxEnt, instead of the notation *f_i*, we will often use the notation *f_i*(c,x), meaning that a feature i **for a particular class c** for a given observation x

Maximum Entropy Modeling

Assume C = {NN, VB}

		f1	f2	f3	f4	f5	f6
VB	f	0	1	0	1	1	0
VB	w		.8		.01	.1	
NN	f	1	0	0	0	0	1
NN	w	.8					-1.3

$$P(NN|x) = \frac{e^{.8}e^{-1.3}}{e^{.8}e^{-1.3} + e^{.8}e^{.01}e^{.1}} = .20$$

$$P(VB|x) = \frac{e^{.8}e^{.01}e^{.1}}{e^{.8}e^{-1.3} + e^{.8}e^{.01}e^{.1}} = .80$$

$$\hat{c} = \underset{c \in C}{\operatorname{argmax}} P(c|x)$$

$$f_1(c,x) = \begin{cases} 1 & \text{if } word_i = \text{“race” } \& \ c = \text{NN} \\ 0 & \text{otherwise} \end{cases}$$

$$f_2(c,x) = \begin{cases} 1 & \text{if } t_{i-1} = \text{TO } \& \ c = \text{VB} \\ 0 & \text{otherwise} \end{cases}$$

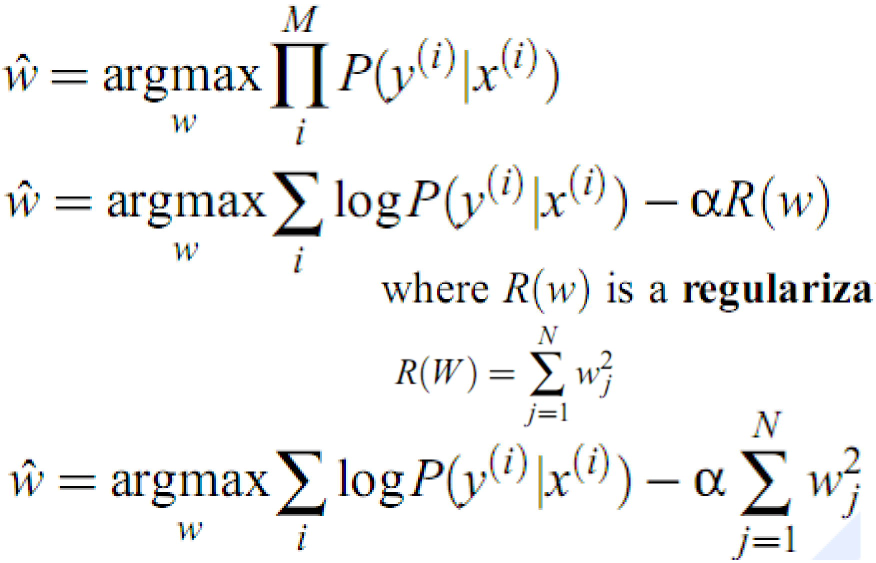
$$f_3(c,x) = \begin{cases} 1 & \text{if suffix}(word_i) = \text{“ing” } \& \ c = \text{VBG} \\ 0 & \text{otherwise} \end{cases}$$

$$f_4(c,x) = \begin{cases} 1 & \text{if is\_lower\_case}(word_i) \ \& \ c = \text{VB} \\ 0 & \text{otherwise} \end{cases}$$

$$f_5(c,x) = \begin{cases} 1 & \text{if } word_i = \text{”race” } \& \ c = \text{VB} \\ 0 & \text{otherwise} \end{cases}$$

$$f_6(c,x) = \begin{cases} 1 & \text{if } t_{i-1} = \text{TO } \& \ c = \text{NN} \\ 0 & \text{otherwise} \end{cases}$$

Learning Maximum Entropy Model



$$\hat{w} = \underset{w}{\operatorname{argmax}} \prod_{i}^{M} P(y^{(i)}|x^{(i)})$$

$$\hat{w} = \underset{w}{\operatorname{argmax}} \sum_{i} \log P(y^{(i)}|x^{(i)}) - \alpha R(w)$$

where $R(w)$ is a **regularization** term used to penalize large weights

$$R(W) = \sum_{j=1}^{N} w_j^2$$

$$\hat{w} = \underset{w}{\operatorname{argmax}} \sum_{i} \log P(y^{(i)}|x^{(i)}) - \alpha \sum_{j=1}^{N} w_j^2$$

Why the name "MaxEnt"?

Tag the word "*zzfish*"

1. Take fewest assumption, imposing no constraint

NN	JJ	NNS	VB	NNP	IN	MD	UH	SYM	VBG	…
1/45	1/45	1/45	1/45	1/45	1/45	1/45	1/45	1/45	1/45	…

Entropy
?

2. Set of possible tags: NN, JJ, NNS, VB

NN	JJ	NNS	VB	NNP	IN	MD	UH	SYM	VBG	…
¼	¼	¼	¼	0	0	0	0	0	0	…

Entropy
?

Why the name "MaxEnt"?

3. 8 times out of 10, *zzfish* was tagged as some sort of common noun (NN/NNS)

NN	JJ	NNS	VB	NNP	IN	MD	UH	SYM	VBG	…
4/10	1/10	4/10	1/10	0	0	0	0	0	0	…

4. For all words, VB occurs as 1 word in 20

NN	JJ	NNS	VB	NNP	IN	MD	UH	SYM	VBG	…
4/10	3/20	4/10	1/20	0	0	0	0	0	0	…

MaxEnt intuition: **Occam's Razor** -- among competing hypotheses, the one with the fewest assumptions should be selected

In Summary

- The intuition of MaxEnt is to build a distribution by continuously adding features

- Each feature is an indicator function, which selects a subset of training observations

- For each feature, add a constraint on our total distribution, specifying that our distribution for this subset should match the empirical distribution of the training set

- We choose the maximum entropy distribution that otherwise accords with these constraints.

Why the name “MaxEnt”?

To select a model from a set C of allowed probability distribution, choose the model p with maximum entropy H(p)*   ---- Berger et al. (1996)

Berger et al. (1996) proved that the solution of this optimization problem is exactly the probability distribution of a multinomial logistic regression whose weights W maximize the likelihood of the training data.

The multinomial logistic regression, when trained according to the maximum likelihood criteria, also finds the maximum entropy distribution subject to the constraints from the feature functions.

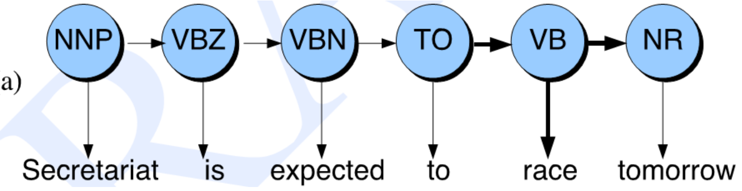
HMM vs. MEMM

MEMM can condition on any useful feature of the input observation; in HMM this isn't possible

HMM

$$\begin{aligned}\hat{T} &= \operatorname*{argmax}_T P(T|W)\\ &= \operatorname*{argmax}_T P(W|T)P(T)\\ &= \operatorname*{argmax}_T \prod_i P(word_i|tag_i) \prod_i P(tag_i|tag_{i-1})\end{aligned}$$

(a)



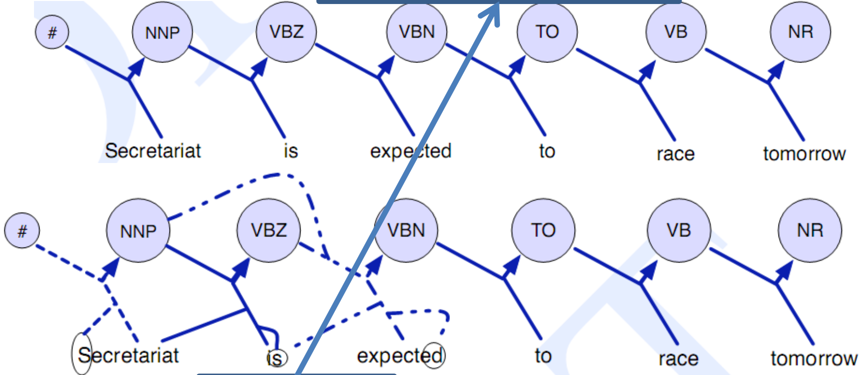
$$P(Q|O) = \prod_{i=1}^{n} P(o_i|q_i) \times \prod_{i=1}^{n} P(q_i|q_{i-1})$$

$$v_t(j) = \max_{1\leq i\leq N-1} v_{t-1}(i)\, \underline{P(s_j|s_i)\, P(o_t|s_j)}$$

$$v_t(j) = \max_{1\leq i\leq N-1} v_{t-1}(i)\, \underline{a_{ij}\, b_j(o_t)}; \quad 1 < j < N, 1 < t < T$$

MEMM

$$\begin{aligned}\hat{T} &= \operatorname*{argmax}_T P(T|W)\\ &= \operatorname*{argmax}_T \prod_i \boxed{P(tag_i|word_i, tag_{i-1})}\end{aligned}$$

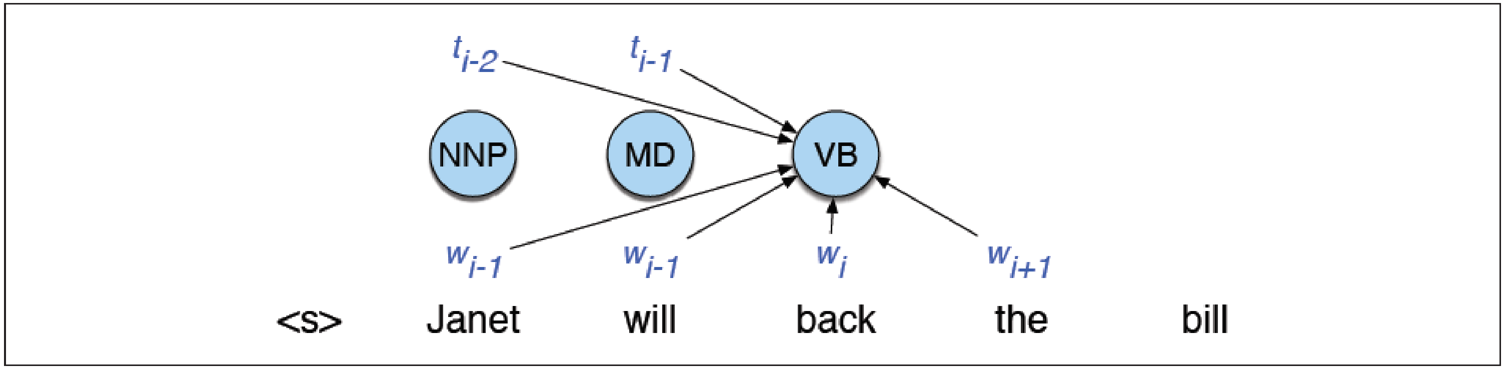


$$P(Q|O) = \prod_{i=1}^{n} \boxed{P(q_i|q_{i-1}, o_i)}$$

$$\boxed{P(q|q', o)} = \frac{1}{Z(o,q')} \exp\left(\sum_i w_i f_i(\overset{\text{word}}{o}, \underset{\text{class}}{q})\right)$$

$$v_t(j) = \max_{1\leq i\leq N-1} v_{t-1}(i)\, \underline{P(s_j|s_i, o_t)}$$

Features in MEMM

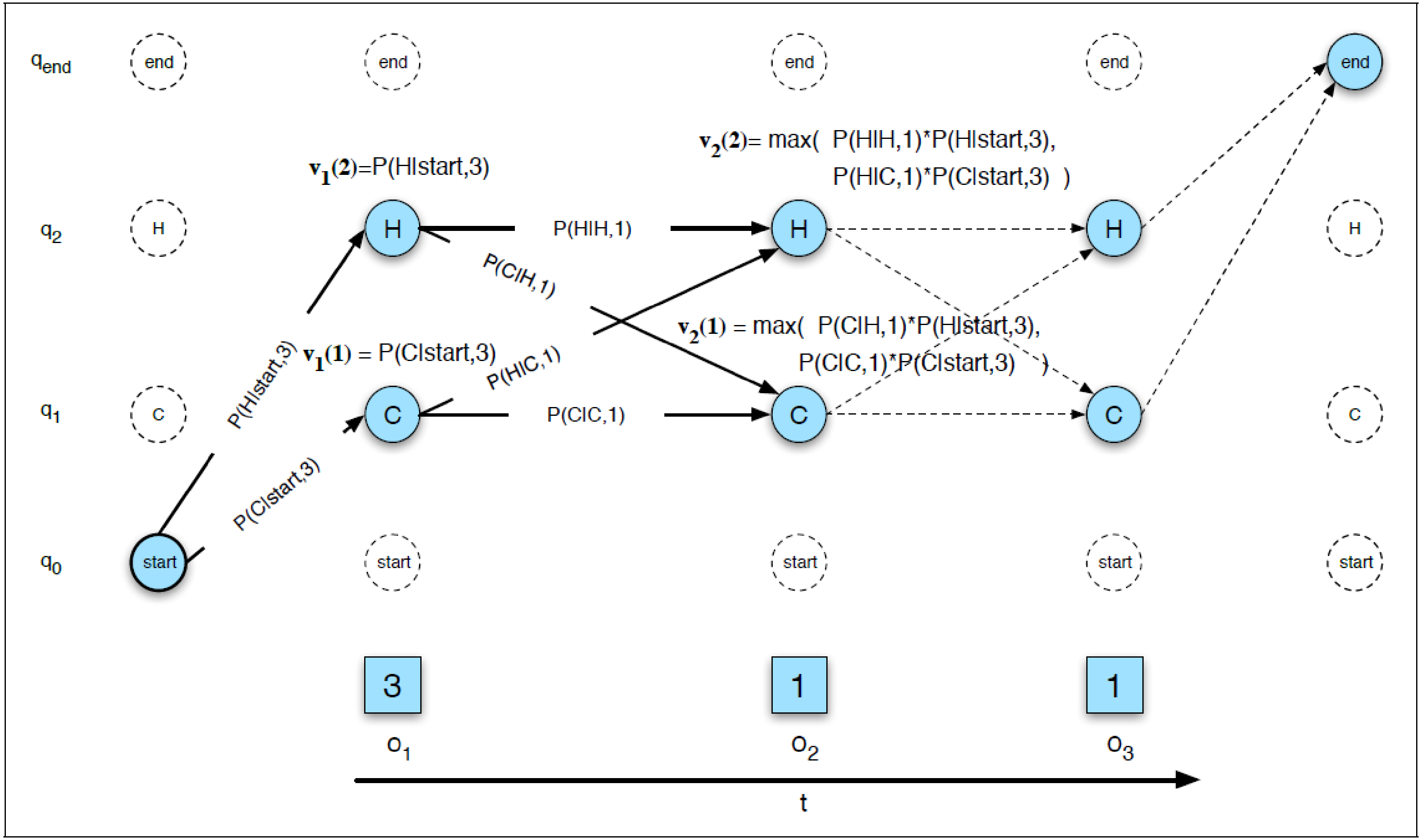


Feature Template

$$\langle t_i, w_{i-2}\rangle, \langle t_i, w_{i-1}\rangle, \langle t_i, w_i\rangle, \langle t_i, w_{i+1}\rangle, \langle t_i, w_{i+2}\rangle$$
$$\langle t_i, t_{i-1}\rangle, \langle t_i, t_{i-2}, t_{i-1}\rangle,$$
$$\langle t_i, t_{i-1}, w_i\rangle, \langle t_i, w_{i-1}, w_i\rangle \langle t_i, w_i, w_{i+1}\rangle,$$

$$\begin{aligned}
\hat{T} &= \underset{T}{\operatorname{argmax}} P(T|W) \\
&= \underset{T}{\operatorname{argmax}} \prod_i P(t_i | w_{i-l}^{i+l}, t_{i-k}^{i-1}) \\
&= \underset{T}{\operatorname{argmax}} \prod_i \frac{\exp\left(\sum_i w_i f_i(t_i, w_{i-l}^{i+l}, t_{i-k}^{i-1})\right)}{\sum_{t' \in \text{tagset}} \exp\left(\sum_i w_i f_i(t', w_{i-l}^{i+l}, t_{i-k}^{i-1})\right)}
\end{aligned}$$

Decoding and Training MEMMs



q_{end}
q_2
q_1
q_0
end
end
end
end
end
H
H
H
H
H
C
C
C
C
C
start
start
start
start
start
$v_1(2)$=P(H|start,3)
$v_1(1)$ = P(C|start,3)
$v_2(2)$= max(P(H|H,1)*P(H|start,3),
P(H|C,1)*P(C|start,3))
$v_2(1)$ = max(P(C|H,1)*P(H|start,3),
P(C|C,1)*P(C|start,3))
P(H|start,3)
P(C|start,3)
P(H|H,1)
P(C|H,1)
P(H|C,1)
P(C|C,1)
3
1
1
o_1
o_2
o_3
t