Tanulás: rejtett Markov-modellek és maximum entrópia Markov-modellek

Az óra anyaga az alábbi könyv 5. és 6. fejezetére támaszkodik:

Jurafsky, Daniel, and James H. Martin. 2009. Speech and Language Processing: An Introduction to Natural Language Processing, Speech Recognition, and Computational Linguistics. 2nd edition. Prentice-Hall.

Gépi tanuló modellek

- **Adatok eloszlásának matematikai közelítése**
- **Találjunk olyan modelleket, amelyek jól leírják az adatokat**
- **Pl. jól szeparálják a pozitív és negatív példákat**
- **Meg tudják jósolni az új példák címkéit**

Token vagy szekvencia?

- **Mi egy egység?**
- **1 esemény vs. eseménysorozat**

a1, a2, a3…. an

- **NLP: 1 szó vs. 1 mondat**
- **Tokenalapú megközelítések: minden alapegység külön címkéződik**
- **Szekvenciajelölők: egyszerre címkézzük fel a teljes szekvenciát (HMM és MEMM)**

Markov-láncok

- **Megfigyelt Markov-modell**
- **Állapotok**
- **Valószínűségek**
- **Kezdő- és végállapot**
- **A bemenet egyértelműen meghatározza a következő állapotot**
- **Adott állapot valószínűsége csak az előző állapottól függ**

$Q = q_1 q_2 \ldots q_N$	a set of N **states**
$A = a_{01} a_{02} \ldots a_{n1} \ldots a_{nn}$	a **transition probability matrix** A, each a_{ij} representing the probability of moving from state i to state j, s.t. $\sum_{j=1}^{n} a_{ij} = 1 \quad \forall i$
q_0, q_F	a special **start state** and **end (final) state** which are not associated with observations.

Markov Assumption: $P(q_i | q_1 \ldots q_{i-1}) = P(q_i | q_{i-1})$

Egy példa

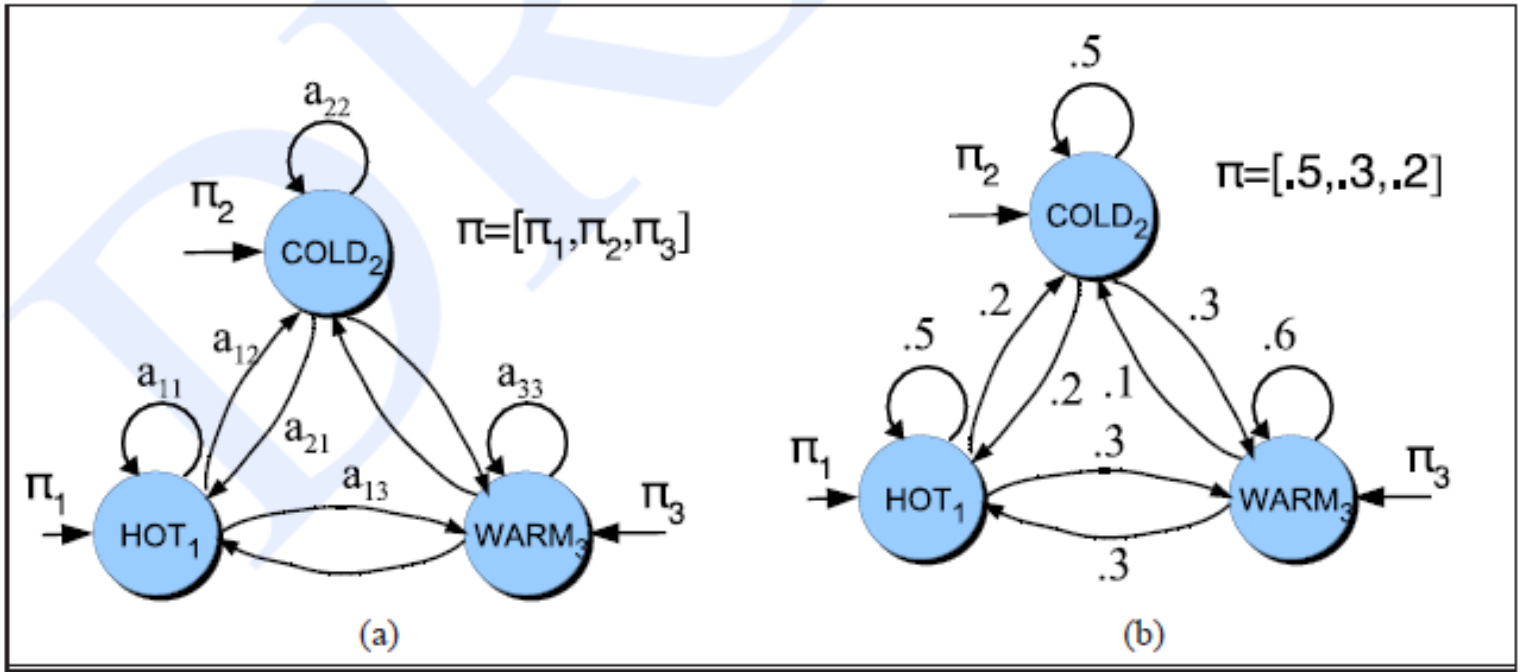


Figure 6.2 Another representation of the same Markov chain for weather shown in Fig. 6.1. Instead of using a special start state with a_{01} transition probabilities, we use the π vector, which represents the distribution over starting state probabilities. The figure in (b) shows sample probabilities.

Rejtett Markov-modell

- **Nem minden figyelhető meg a világban**
- **Rejtett változók**
- **Megfigyelt események**
- **Rejtett események**
- **Példa: elfogyasztott fagyik száma (megfigyelt) + időjárás (rejtett)**

$Q = q_1 q_2 \ldots q_N$	a set of N **states**
$A = a_{11} a_{12} \ldots a_{n1} \ldots a_{nn}$	a **transition probability matrix** A, each a_{ij} representing the probability of moving from state i to state j, s.t. $\sum_{j=1}^{n} a_{ij} = 1 \quad \forall i$
$O = o_1 o_2 \ldots o_T$	a sequence of T **observations**, each one drawn from a vocabulary $V = v_1, v_2, ..., v_V$.
$B = b_i(o_t)$	a sequence of **observation likelihoods**:, also called **emission probabilities**, each expressing the probability of an observation o_t being generated from a state i.
q_0, q_F	a special **start state** and **end (final) state** which are not associated with observations, together with transition probabilities $a_{01} a_{02} .. a_{0n}$ out of the start state and $a_{1F} a_{2F} ... a_{nF}$ into the end state.

Feltevések

- **Adott állapot valószínűsége csak az előző állapottól függ**
- **A megfigyelés valószínűsége csak a megfigyelést kiváltó állapottól függ**

Output Independence Assumption: $P(o_i|q_1 \ldots q_i, \ldots, q_T, o_1, \ldots, o_i, \ldots, o_T) = P(o_i|q_i)$

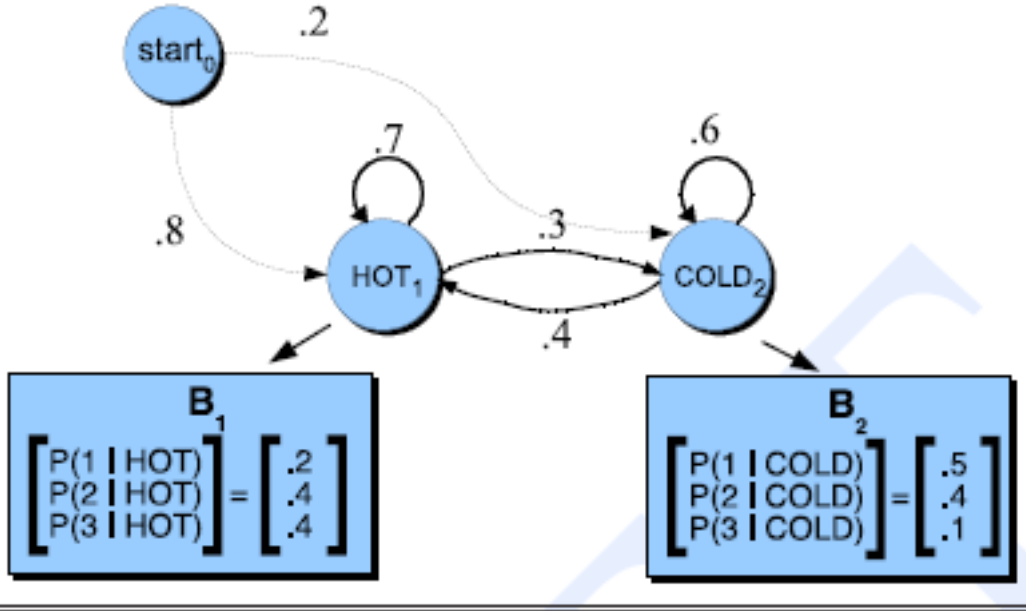


Figure 6.3 A Hidden Markov Model for relating numbers of ice creams eaten by Jason (the observations) to the weather (H or C, the hidden variables). For this example we are not using an end-state, instead allowing both states 1 and 2 to be a final (accepting) state.

Alapproblémák

- **A valószínűségek kiszámítása**
- **Dekódolás (a rejtett állapotok megtalálása)**
- **Tanulás (a megfigyelések és az állapotok alapján határozzuk meg az átmeneti és a megfigyelési valószínűségeket)**

A valószínűségek kiszámítása

- **Minden rejtett állapothoz 1 megfigyelés tartozik**
- **Megfigyelésből következtetünk vissza**
- **Ez lenne a cél…**

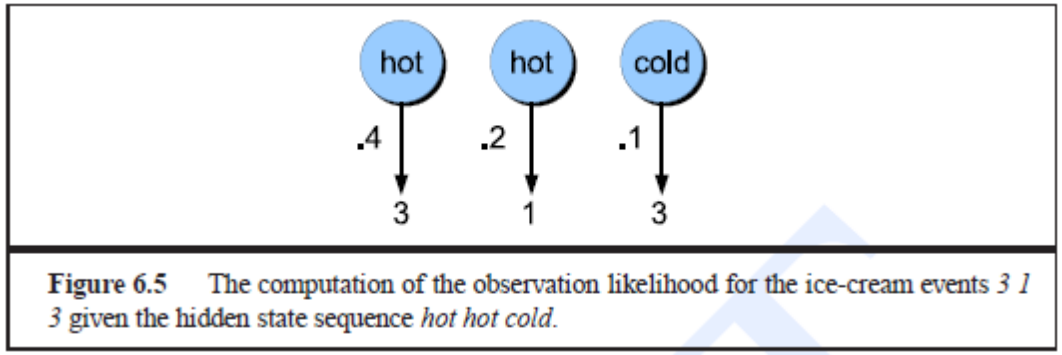


Figure 6.5 The computation of the observation likelihood for the ice-cream events *3 1 3* given the hidden state sequence *hot hot cold*.

Együttes valószínűségek

- **Q időjárási szekvencia és az általa kiváltott O fagyizási események együttes valószínűsége**
- **Megnézzük minden lehetséges rejtett (időjárási) szekvenciára**

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

$$\begin{aligned} P(3\ 1\ 3, \text{hot hot cold}) &= P(\text{hot}|\text{start}) \times P(\text{hot}|\text{hot}) \times P(\text{cold}|\text{hot}) \\ &\quad \times P(3|\text{hot}) \times P(1|\text{hot}) \times P(3|\text{cold}) \end{aligned}$$

$$P(O) = \sum_{Q} P(O,Q) = \sum_{Q} P(O|Q)P(Q)$$

$$P(3\ 1\ 3) = P(3\ 1\ 3, \text{cold cold cold}) + P(3\ 1\ 3, \text{cold cold hot}) + P(3\ 1\ 3, \text{hot hot cold}) + \ldots$$

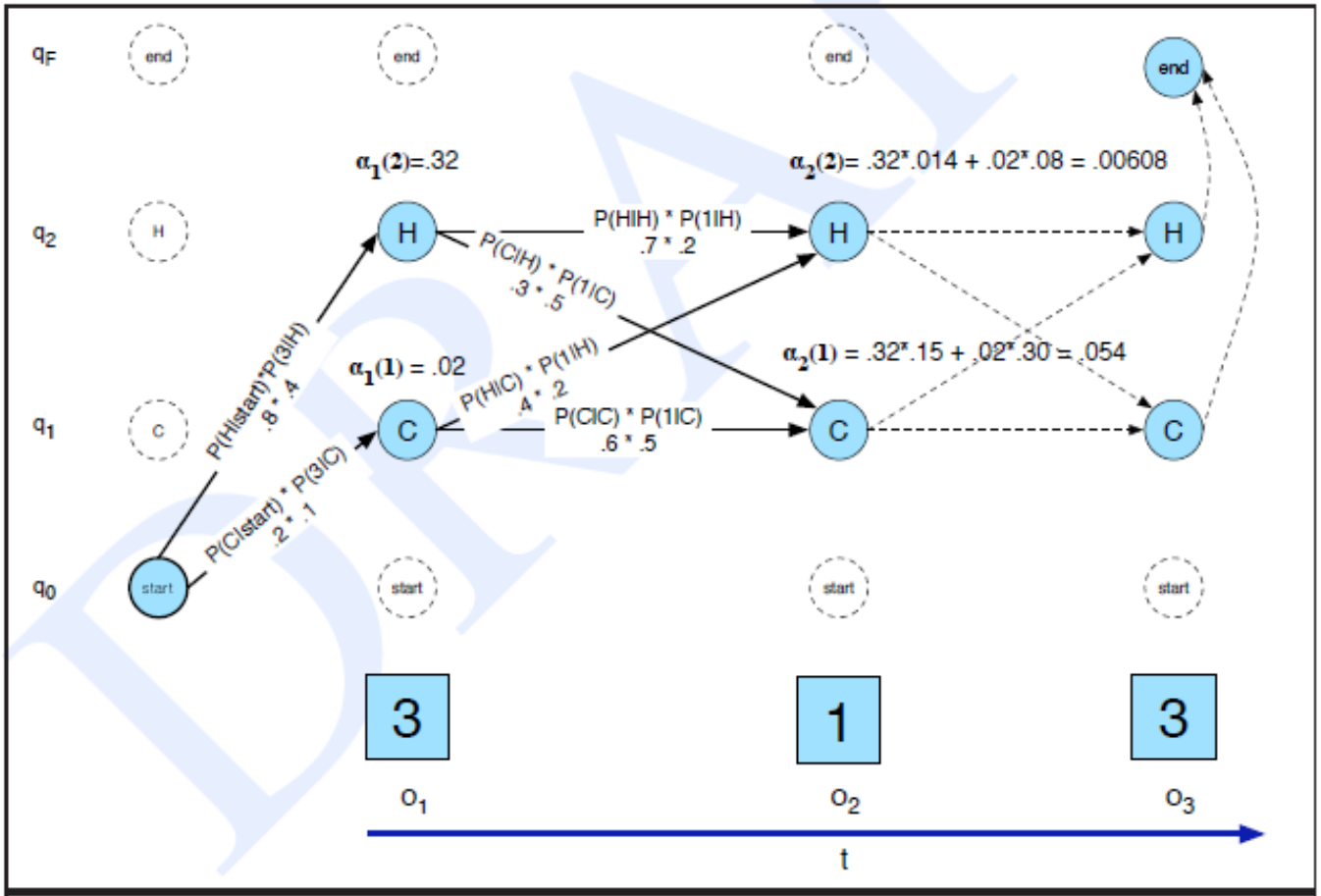
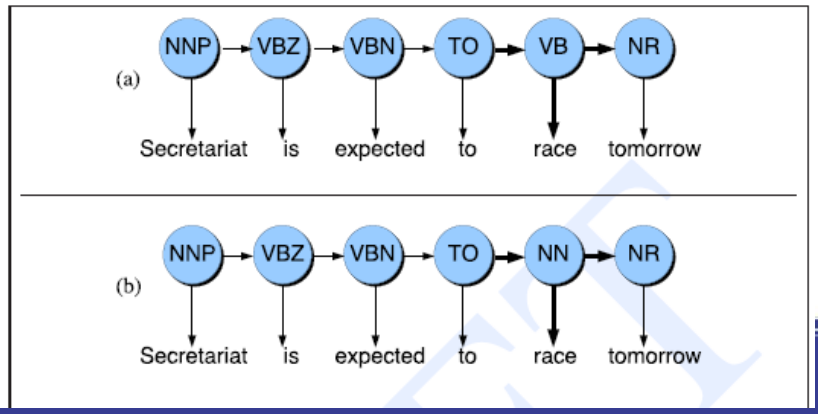


Figure 6.7 The forward trellis for computing the total observation likelihood for the ice-cream events *3 1 3*. Hidden states are in circles, observations in squares. White (unfilled) circles indicate illegal transitions. The figure shows the computation of $\alpha_t(j)$ for two states at two time steps. The computation in each cell follows Eq˙ 6.15: $\alpha_t(j) = \sum_{i=1}^{N} \alpha_{t-1}(i) a_{ij} b_j(o_t)$. The resulting probability expressed in each cell is Eq˙ 6.14: $\alpha_t(j) = P(o_1, o_2 \ldots o_t, q_t = j|\lambda)$.

Szófaji egyértelműsítés (HMM)

- **Minden szóra a legvalószínűbb szófajt választjuk: P(T|W)**
- **Bayes-szabály alkalmazásával felbontjuk: P(W|T) P(T)**
- **Feltételezés: a szó előfordulási valószínűsége csak a címkétől függ + címke valószínűsége csak a megelőző címkétől függ**
- **Címkeátmenet-valószínűségek (adott címkét milyen másik követ?)**
- **P(Wn|Tn) P(Tn|Tn-1)**



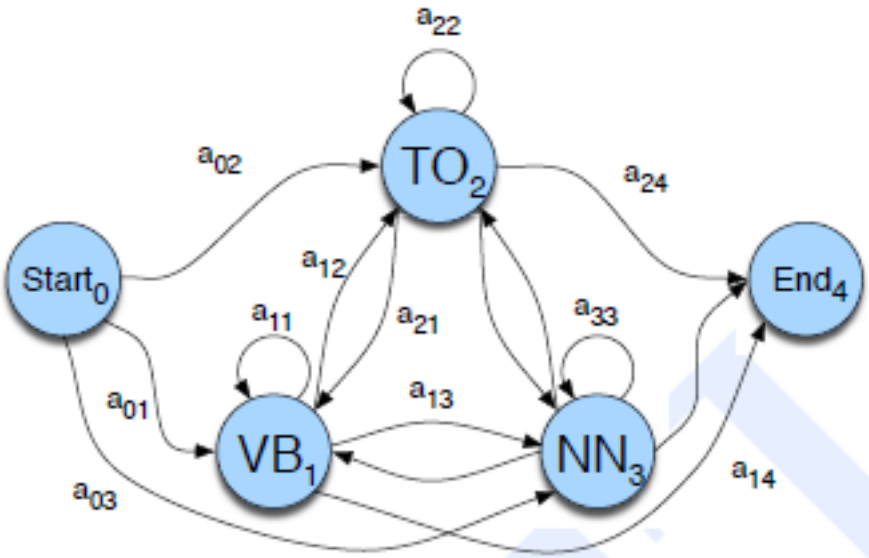


Figure 5.13 The Markov chain corresponding to the hidden states of the HMM. The *A* transition probabilities are used to compute the prior probability.

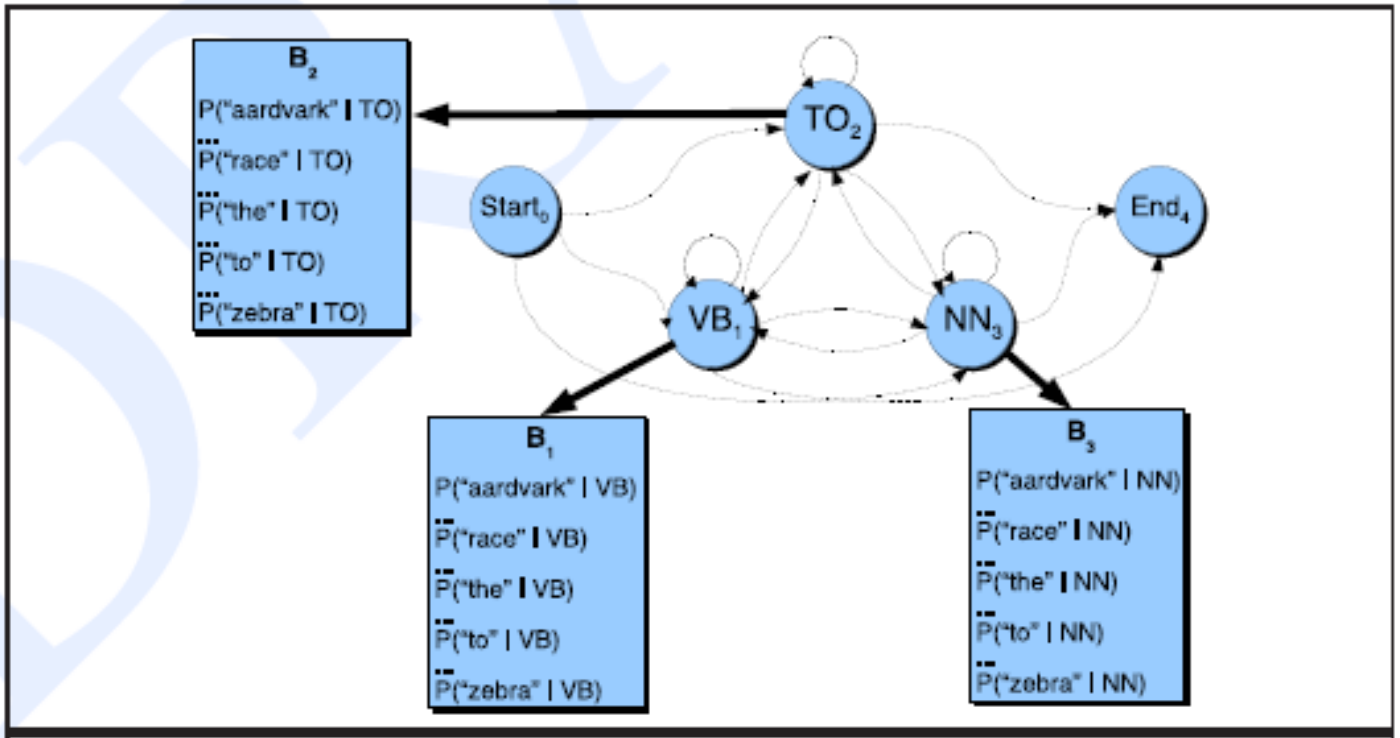


Figure 5.14 The *B* observation likelihoods for the HMM in the previous figure. Each state (except the non-emitting Start and End states) is associated with a vector of probabilities, one likelihood for each possible observation word.

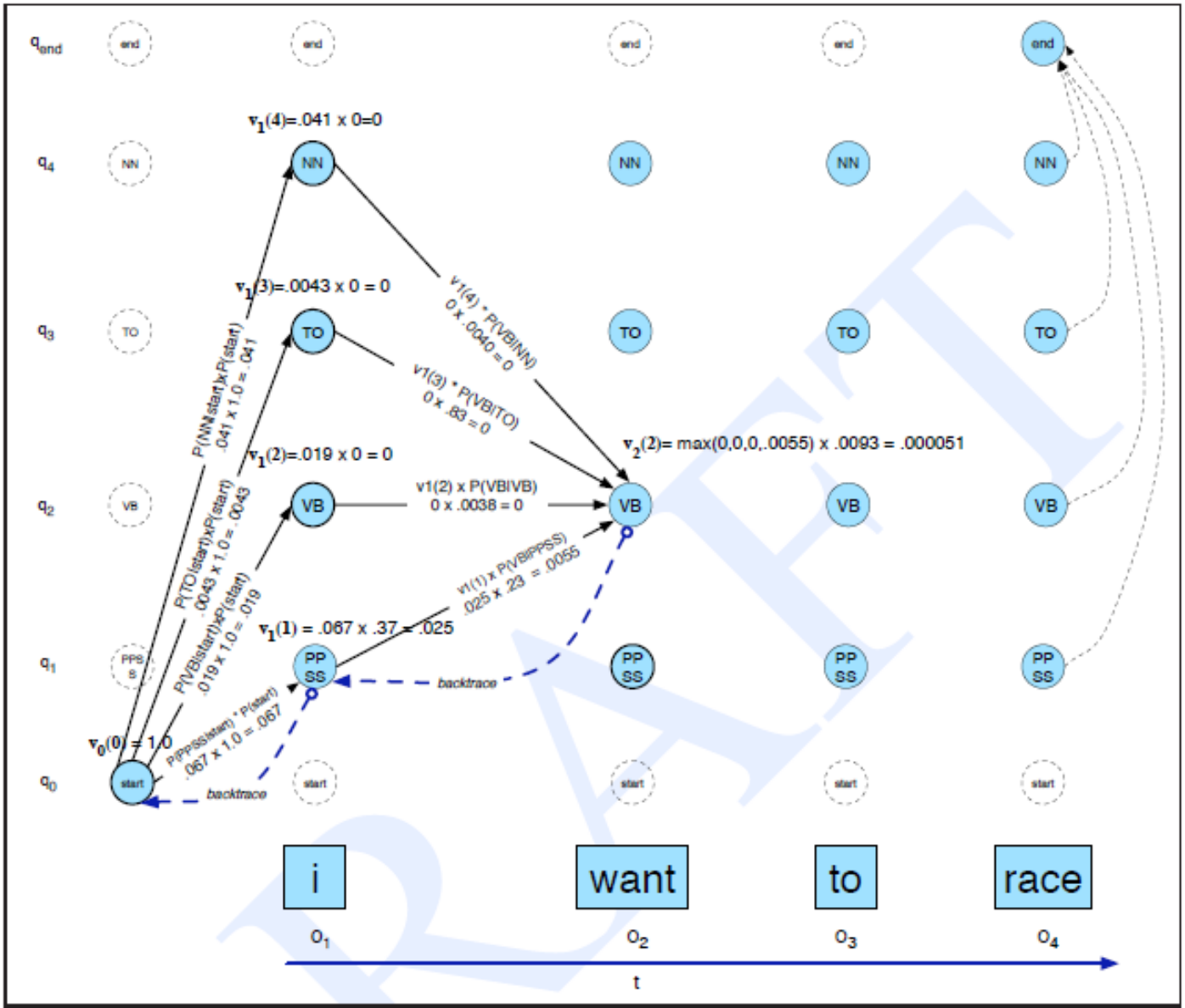


Figure 5.18 The entries in the individual state columns for the Viterbi algorithm. Each cell keeps the probability of the best path so far and a pointer to the previous cell along that path. We have only filled out columns 0 and 1 and one cell of column 2; the rest is left as an exercise for the reader. After the cells are filled in, backtracing from the *end* state, we should be able to reconstruct the correct state sequence PPSS VB TO VB.

Lineáris regresszió

- **Folytonos értékek**
- **Ingatlanhirdetésekben található szubjektív melléknevek száma – eladási ár**

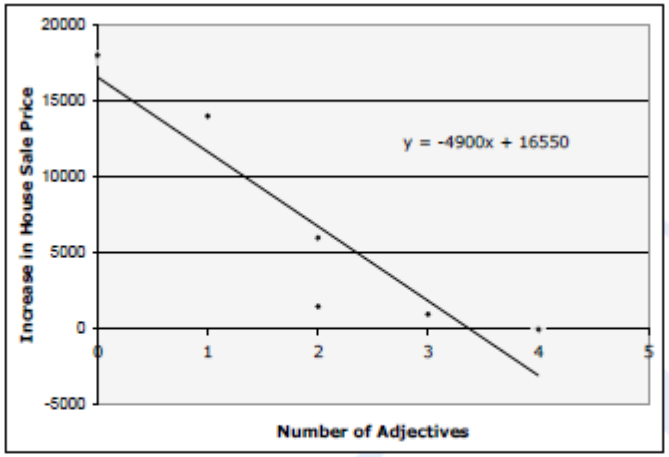


Figure 6.18 A plot of the (made-up) points in Fig. 6.17 and the regression line that best fits them, with the equation $y = -4900x + 16550$.

Többszörös lineáris regresszió

- **Több jellemzőt is figyelembe veszünk (pl. törlesztőrészlet nagysága, eladásra váró házak száma…)**
- **Súlyozzuk őket (mi a fontosabb? Mi számít jobban?)**

Logisztikus regresszió

- **Diszkrét változók ~ osztályozás**
- **Valószínűségeket ad vissza (mekkora valószínűséggel tartozik az adott osztályba) – „odds”**
- **Természetes alapú logaritmus alapján**
- **1 / (1 + e exp –x)**

MaxEnt

- **Maximum entrópia**
- **Logisztikus regresszió sok diszkrét értékre**
- **C: osztály, f: jellemző, x: vizsgált egyed**

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

Példa: szófaji egyértelműsítés

Secretariat/NNP is/BEZ expected/VBN to/TO race/??? tomorrow/RB

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

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

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

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

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

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

		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

Figure 6.19 Some sample feature values and weights for tagging the word *race* in (6.81).

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

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

Maximum entrópia

- **Entrópia: rendezetlenség**
- **Maximális entrópia: minden címkének ugyanakkora a valószínűsége, ha nincs egyéb infó**
- **DE: ha beépítjük a (tanuló halmazon tett) megfigyeléseinket, akkor megszorítások is lesznek a jellemzőkre nézve**
- **A megszorításoknak megfelelő legvalószínűbb címkét választjuk**
- **MaxEnt: megszorításokkal összhangban levő maximális entrópia megtalálása**

Maximum entrópia Markov-modell (MEMM)

- **MaxEnt önmagában 1 elem besorolására képes**
- **Nem szekvenciákkal dolgozik**
- **Bővítjük a modellt -> MEMM**

Szófaji egyértelműsítés (MEMM)

- **W: szó, T: címke**
- **P(T|W) = ??**
- **HMM: Bayes-szabály alapján P(W|T)P(W)-t számít**
- **MEMM: P(T|W)-t közvetlenül számítja előző címke és a szó alapján**

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

- **Diszkriminatív modell: lehetséges szófaji szekvenciák között tesz különbséget**

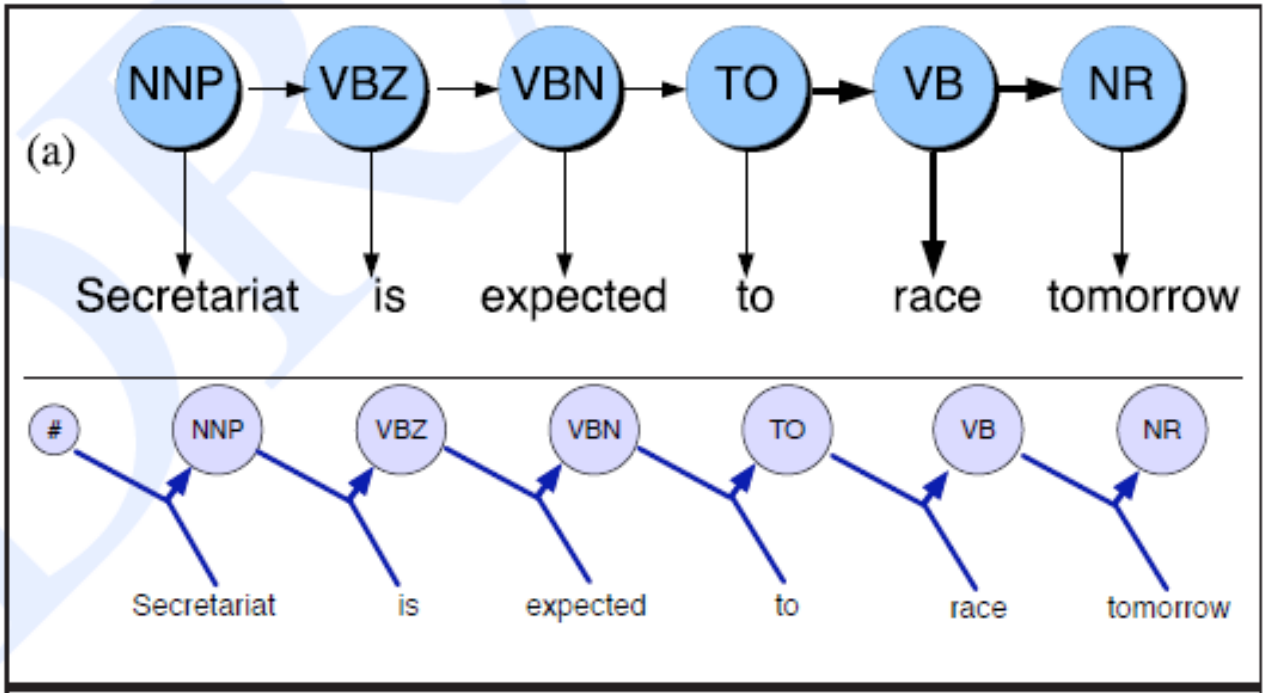


Figure 6.20 The HMM (top) and MEMM (bottom) representation of the probability computation for the correct sequence of tags for the Secretariat sentence. Each arc would be associated with a probability; the HMM computes two separate probabilities for the observation likelihood and the prior, while the MEMM computes a single probability function at each state, conditioned on the previous state and current observation.

Egyéb jellemzők

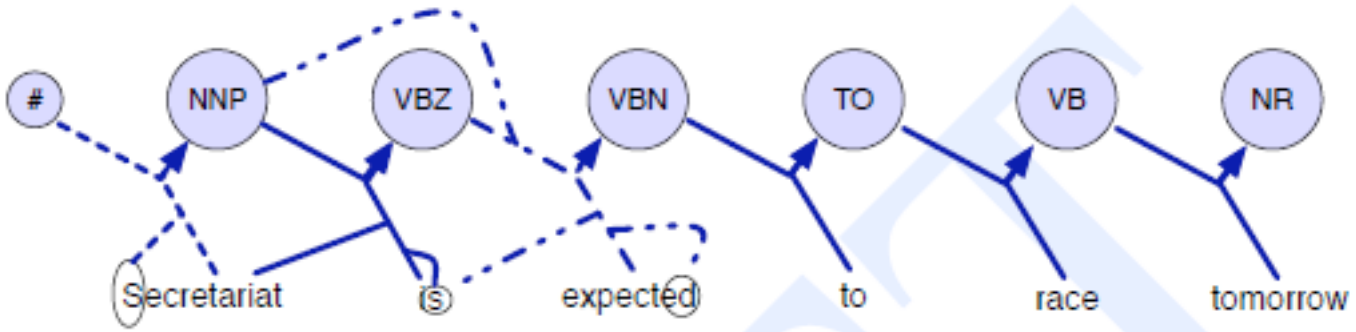


Figure 6.21 An MEMM for part-of-speech tagging, augmenting the description in Fig. 6.20 by showing that an MEMM can condition on many features of the input, such as capitalization, morphology (ending in *-s* or *-ed*), as well as earlier words or tags. We have shown some potential additional features for the first three decisions, using different line styles for each class.

HMM vs. MEMM

- **HMM:**

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

- **MEMM:**

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

- **HMM-nél sok valószínűséget ismernünk kellene (minden jellemzőre ki kéne számolni)**
- **A gyakorlatban több jellemzőt is képes kezelni a MEMM**