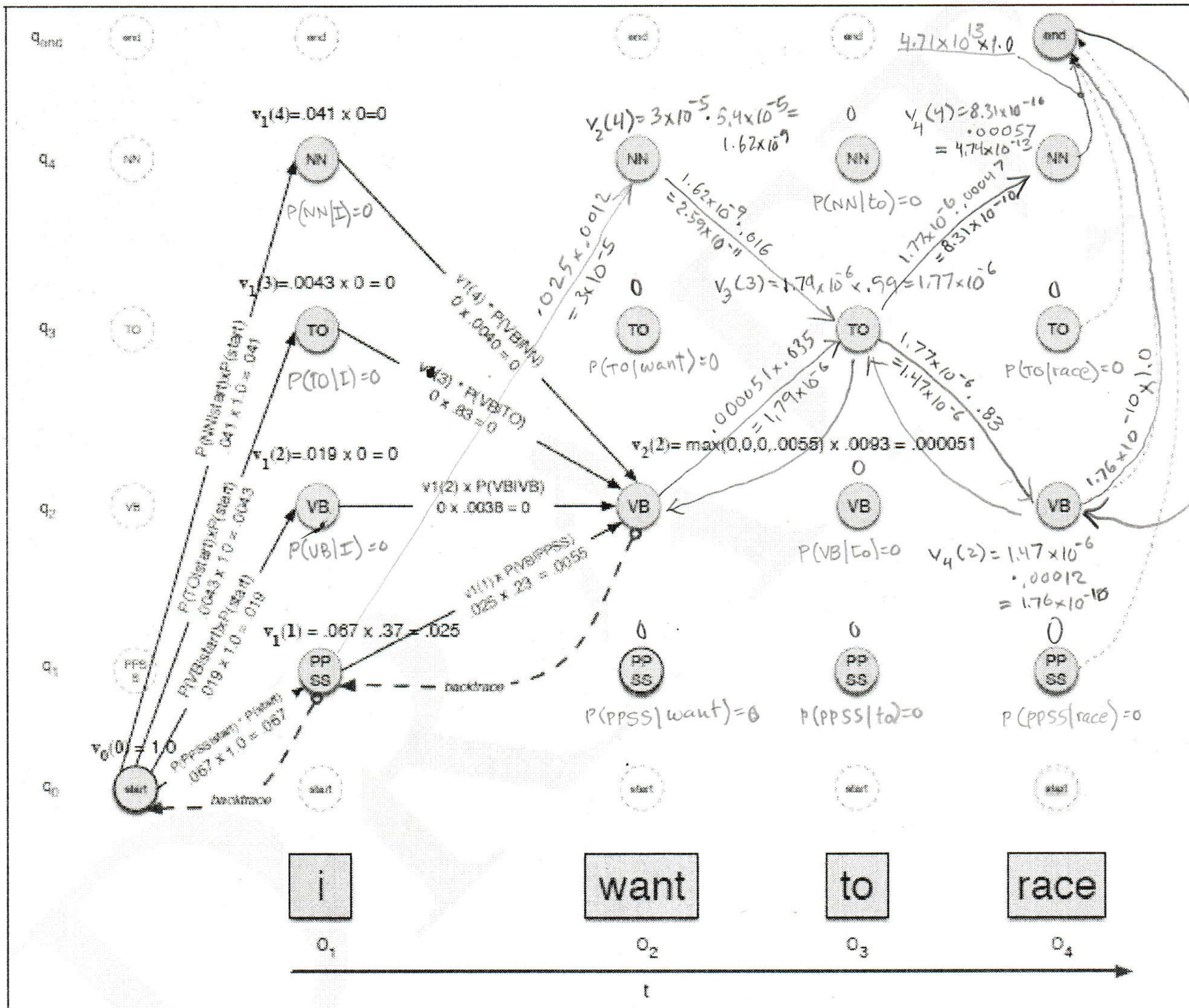

Figure 5.18 The entries in the individual state columns for the Viterbi algorithm. Each cell keeps the probability of the best path so far and a pointer to the previous cell along that path. We have only filled out columns 0 and 1 and one cell of column 2; the rest is left as an exercise for the reader. 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.

Don't consider paths from/to nodes with $b_s(o_t) = 0$ e.g. P(TO|want) = 0
Don't consider paths from nodes with $v_t(s) = 0$ or P(Target|Source) = 0

Final result is: i/PPSS want/VB to/TO race/VB

	VB	TO	NN	PPSS
<s>	.019	.0043	.041	.067
VB	.0038	.035	.047	.0070
TO	.83	0	.00047	0
NN	.0040	.016	.087	.0045
PPSS	.23	.00079	.0012	.00014

Figure 5.15 Tag transition probabilities (the a array, $p(t_i|t_{i-1})$) computed from the 87-tag Brown corpus without smoothing. The rows are labeled with the conditioning event; thus $P(PPSS|VB)$ is .0070. The symbol <s> is the start-of-sentence symbol.

	I	want	to	race
VB	0	.0093	0	.00012
TO	0	0	.99	0
NN	0	.000054	0	.00057
PPSS	.37	0	0	0

Figure 5.16 Observation likelihoods (the b array) computed from the 87-tag Brown corpus without smoothing.

function VITERBI(*observations* of len *T*,*state-graph* of len *N*) **returns** *best-path*

create a path probability matrix *viterbi[N+2,T]*
for each state *s* **from** 1 **to** *N* **do** ;initialization step
  $viterbi[s,1] \leftarrow a_{0,s} * b_s(o_1)$
  $backpointer[s,1] \leftarrow 0$
for each time step *t* **from** 2 **to** *T* **do** ;recursion step
  for each state *s* **from** 1 **to** *N* **do**
    $viterbi[s,t] \leftarrow \max_{s'=1}^{N} viterbi[s',t-1] * a_{s',s} * b_s(o_t)$
    $backpointer[s,t] \leftarrow \operatorname{argmax}_{s'=1}^{N} viterbi[s',t-1] * a_{s',s}$
$viterbi[q_F,T] \leftarrow \max_{s=1}^{N} viterbi[s,T] * a_{s,q_F}$; termination step
$backpointer[q_F,T] \leftarrow \operatorname{argmax}_{s=1}^{N} viterbi[s,T] * a_{s,q_F}$; termination step
  return the backtrace path by following backpointers to states back in time from $backpointer[q_F,T]$

Figure 5.17 Viterbi algorithm for finding optimal sequence of tags. Given an observation sequence and an HMM $\lambda = (A,B)$, the algorithm returns the state-path through the HMM which assigns maximum likelihood to the observation sequence. Note that states 0 and q_F are non-emitting.

30 Aug 2018

Figure 5.18 Viterbi algorithm example

o_2: want

NN: from $PPSS_1$: $.025 \times .0012 = 3 \times 10^{-5}$

TO: from $PPSS_1$: $.025 \times .00079 = 1.975 \times 10^{-5}$

* VB: from $PPSS_1$: $.025 \times .23 = 5.75 \times 10^{-3}$

PPSS: from $PPSS_1$: $.025 \times .00014 = 3.5 \times 10^{-6}$

NN: $v_2(4) = .00003 \times .000054 = 1.62 \times 10^{-9}$

TO: $v_2(3) = 1.975 \times 10^{-5} \times 0 = 0$

* VB: $v_2(2) = 5.75 \times 10^{-3} \times .0093 = 5.35 \times 10^{-5}$

PPSS: $v_2(1) = 3.5 \times 10^{-6} \times 0 = 0$

o_3: to

NN: from NN_2: $1.62 \times 10^{-9} \cdot .087 = 1.41 \times 10^{-10}$

from VB_2: $5.35 \times 10^{-5} \cdot .047 = 2.51 \times 10^{-6}$

TO: from NN_2: $1.62 \times 10^{-9} \cdot .016 = 2.59 \times 10^{-11}$

* from VB_2: $5.35 \times 10^{-5} \cdot .035 = 1.87 \times 10^{-6}$

VB: from NN_2: $1.62 \times 10^{-9} \cdot .0040 = 6.48 \times 10^{-12}$

from VB_2: $5.35 \times 10^{-5} \cdot .0038 = 2.03 \times 10^{-7}$

PPSS: from NN_2: $1.62 \times 10^{-9} \cdot .0045 = 7.29 \times 10^{-12}$

from VB_2: $5.35 \times 10^{-5} \cdot .0070 = 3.74 \times 10^{-7}$

$NN_3: \ v_3(4) = 2.51\times10^{-6} \cdot 0 = 0$

* $TO_3: \ v_3(3) = 1.87\times10^{-6} \cdot .99 = 1.85\times10^{-6}$

$VB_3: \ v_3(2) = 2.03\times10^{-7} \cdot 0 = 0$

$PPSS_3: \ v_3(1) = 3.74\times10^{-7} \cdot 0 = 0$

o_4: race

NN: from TO_3: $1.85\times10^{-6} \cdot .00047 = 8.71\times10^{-10}$

TO: from TO_3: $1.85\times10^{-6} \cdot 0 = 0$

* VB: from TO_3: $1.85\times10^{-6} \cdot .83 = 1.54\times10^{-6}$

PPSS: from TO_3: $1.85\times10^{-6} \cdot 0 = 0$

NN: $v_4(4) = 8.71\times10^{-10} \cdot .00057 = 4.96\times10^{-13}$

TO: $v_4(3) = 0 \cdot 0 = 0$

* VB: $v_4(2) = 1.54\times10^{-6} \cdot .00012 = 1.85\times10^{-10}$

PPSS: $v_4(1) = 0 \cdot 0 = 0$