A Phrase-Based HMM

Hal Daumé III

18 December 2002

1 Introduction

A standard hidden Markov model is represented as a graph, with probabilities on the edges. As edges are transitioned, observable emissions are made. However, the actual path taken in the model is unknown (hence the "hidden" in the name). One common extension to the HMM model is the epsilon transition: allowing transitions to be made without emitting any symbol at all. Here, we present a further, related extension:

Edges may emit more than one observation: that is, they can emit a sequence of observations.

Before we begin, we make some notation:

S	$=$	$\{s_1 \dots s_N\}$	set of states
K	$=$	$\{k_1 \dots k_M\}$	output alphabet
Π	$=$	$\{\pi_j : j \in S\}$	initial state probabilities
A	$=$	$\{a_{i,j} : i, j \in S\}$	transition probabilities
B	$=$	$\{b_{i,j,\bar{k}} : i, j \in S, \bar{k} \in K^+\}$	emission probabilities

Here, the states are indexed by natural numbers from 1 to N (hence there are N states, total). The observation sequences come from the alphabet K (of size M). π_j is the probability of beginning in state s_j. The transition probability $a_{i,j}$ is the probability of transitioning from state s_i to state s_j. $b_{i,j,\bar{k}}$ is the probability of emitting (the non-empty) observation sequence $\bar{k}$ while transitioning from state i to state j. Finally, x_t means the state we are in after emitting the first t symbols.

We use standard sequence notation where x_a^b is shorthand for $\langle x_a \dots x_b \rangle$. If $b < a$ then x_a^b refers to the empty sequence.

Given an observation sequence $o_1^T = \langle o_1 \dots o_T \rangle$, our goal is to estimate the probability of this sequence given a model $\mu = (S, K, \Pi, A, B)$. Thus, we are trying to estimate $\Pr\left(o_1^T \mid \mu\right)$. Considering only the final output from the HMM (remember that this may be one or more symbols), we can break this down into:

$$\Pr\left(o_1^T \mid \mu\right) = \sum_{t=1}^{T} \left(\Pr\left(o_1^{t-1} \mid \mu\right) \cdot \Pr\left(o_t^T \mid \mu, o_1^{t-1}\right) \right)$$

The t parameter in the sum represents the first position of the last output sequence. In the case where $t = 1$, the inner product becomes $1 \cdot \Pr\left(o_1^T \mid \mu\right)$ and the machine is outputting the entire sequence in one transition. In the case where $t = T$, the inner product becomes $\Pr\left(o_1^{T-1} \mid \mu\right) \cdot \Pr\left(o_T \mid \mu, o_1^{T-1}\right)$ in which case this is emitting only one observation in the last transition (assuming, as we will for the remainder of this paper, no epsilon transitions). For t somewhere in the middle, the machine is outputting a phrase of length $T - t + 1$.

At this point, we make the standard Markov assumption:

Assumption: The probability of an observation [sequence] at a given state does not depend on the previous states, but only on the state the machine is currently in.

Thus, in the above expression, the second probability simplifies to:

$$\Pr\left(o_t^T \,\middle|\, \mu, x_{t-1}\right)$$

Where, again, x_{t-1} is state the machine was in after outputting the first $t-1$ symbols (note that this does not mean that it has taken $t-1$ steps: all $t-1$ symbols could have been outputted in one step).

2 Forward

Now we are ready to calculate the actual probability of an observation sequence. We present a dynamic programming solution analogous to the forward algorithm for standard HMMs. We define $\alpha_j(t) := \Pr\left(o_1^{t-1}, x_{t-1} = j \,\middle|\, \mu\right)$, for $j \in S$ and $1 \leq t \leq T+1$: the probability of being in state j after emitting the first $t-1$ symbols (in whatever grouping we want).

Assuming we can calculate these values, we can calculate the probability of an entire observation sequence as $\Pr\left(o_1^T \,\middle|\, \mu\right) = \sum_{j \in S} \Pr\left(o_1^T, x_T = j \,\middle|\, \mu\right) = \sum_{j \in S} \alpha_j(T+1)$.

To calculate the α values, we initialize $\alpha_j(1) = \Pr\left(x_0 = j \,\middle|\, \mu\right) := \pi_j$. We then perform dynamic programming recursion to calculate α for $t > 0$ by:

$$\begin{aligned}
\alpha_j(t+1) &= \Pr\left(o_1^t, x_t = j \,\middle|\, \mu\right) \\
&\quad \textbf{\{consider each possible previous observation and} \\
&\quad \textbf{ each possible previous state\}} \\
&\quad \textbf{\{chain rule\}} \\
&= \sum_{t'=0}^{t-1} \sum_{i \in S} \left(\Pr\left(o_1^{t'}, x_{t'} = i \,\middle|\, \mu\right) \cdot \Pr\left(o_{t'+1}^{t}, x_t = j \,\middle|\, o_1^{t'}, x_{t'} = i, \mu\right)\right) \\
&\quad \textbf{\{Markov assumption and definition of } \boldsymbol{\alpha}\textbf{\}} \\
&= \sum_{t'=0}^{t-1} \sum_{i \in S} \left(\alpha_i(t'+1) \cdot \Pr\left(o_{t'+1}^{t}, x_t = j \,\middle|\, x_{t'} = i, \mu\right)\right) \\
&\quad \textbf{\{definitions of } \boldsymbol{a} \textbf{ and } \boldsymbol{b}\textbf{\}} \\
&= \sum_{t'=0}^{t-1} \sum_{i \in S} \left(\alpha_i(t'+1) \cdot a_{i,j} \cdot b_{i,j,o_{t'+1}^{t}} \right)
\end{aligned}$$

3 Backward

Just as we can compute the probability of an observation sequence by moving forward, so can we calculate it by going backward. We define $\beta_i(t) := \Pr\left(o_t^T \,\middle|\, \mu, x_{t-1} = i\right)$ the probability of emitting the sequence o_t^T given that we are starting out in state i (again, defined for $i \in S$ and $1 \leq t \leq T+1$).

Again, assuming we can calulate these values, we can calculate the probability of an entire observation sequence as $\Pr\left(o_1^T \,\middle|\, \mu\right) = \sum_{i \in S} \Pr\left(x_0 = i \,\middle|\, \mu\right) \cdot \Pr\left(o_1^T \,\middle|\, \mu, x_0 = i\right) = \sum_{i \in S} \pi_i \beta_i(1)$.

We initialize $\beta_i(T+1) = 1$ for all $i \in S$ and then the recursive form of β is derived as:

$$
\begin{aligned}
\beta_i(t) &= \Pr\left(o_t^T \,\middle|\, \mu, x_{t-1} = i\right) \\
&\quad \textbf{\{sum over all breaks in } o\textbf{, and all previous states\}} \\
&= \sum_{t'=t}^{T} \sum_{j \in S} \left(\Pr\left(o_t^{t'}, x_{t'} = j \,\middle|\, \mu, x_{t-1} = i\right) \cdot \Pr\left(o_{t'+1}^T \,\middle|\, \mu, x_{t-1} = i, o_t^{t'}, x_{t'} = j\right)\right) \\
&\quad \textbf{\{Markov assumption, definition of } a \textbf{ and } b\textbf{\}} \\
&= \sum_{t'=t}^{T} \sum_{j \in S} \left(a_{i,j} \cdot b_{i,j,o_t^{t'}} \cdot \Pr\left(o_{t'+1}^T \,\middle|\, \mu, x_{t'} = j\right)\right) \\
&\quad \textbf{\{definition of } \beta\textbf{\}} \\
&= \sum_{t'=t}^{T} \sum_{j \in S} \left(a_{i,j} \cdot b_{i,j,o_t^{t'}} \cdot \beta_j(t'+1)\right)
\end{aligned}
$$

Being able to calculate these backwards values enables us to calculate probabilities of being in a particular state after emitting some symbols. We estimate for any t:

$$
\begin{aligned}
& \Pr\left(o_1^T, x_t = i \,\middle|\, \mu\right) \\
& \quad \textbf{\{splitting the observation sequence\}} \\
={}& \Pr\left(o_1^t, x_t = i, o_t^T \,\middle|\, \mu\right) \\
& \quad \textbf{\{chain rule\}} \\
={}& \Pr\left(o_1^t, x_t = i \,\middle|\, \mu\right) \cdot \Pr\left(o_{t+1}^T \,\middle|\, \mu, x_t = i, o_1^t\right) \\
& \quad \textbf{\{Markov assumption\}} \\
={}& \Pr\left(o_1^t, x_t = i \,\middle|\, \mu\right) \cdot \Pr\left(o_{t+1}^T \,\middle|\, \mu, x_t = i\right) \\
& \quad \textbf{\{definitions of } \alpha \textbf{ and } \beta\textbf{\}} \\
={}& \alpha_i(t+1) \cdot \beta_i(t+1)
\end{aligned}
$$

Therefore, we get $\Pr\left(o_1^T \,\middle|\, \mu\right) = \sum_{i \in S} \alpha_i(t)\beta_i(t)$ for any $0 \leq t \leq T$. In particular, when $t = 0$, this becomes

$$
\begin{aligned}
& \sum_{i \in S} (\alpha_i(1)\beta_i(1)) \\
& \quad \textbf{\{definition of } \alpha \textbf{\}} \\
= \; & \sum_{i \in S} (\pi_i \beta_i(1)) \\
& \quad \textbf{\{definition of } \pi \textbf{ and } \beta \textbf{\}} \\
= \; & \sum_{i \in S} \left(\Pr(x_0 = i) \Pr\left(o_1^T \,\middle|\, \mu, x_0 = i\right)\right) \\
& \quad \textbf{\{math\}} \\
= \; & \Pr\left(o_1^T \,\middle|\, \mu\right)
\end{aligned}
$$

We can perform the same calculation when $t = T$ and get:

$$
\begin{aligned}
& \sum_{i \in S} (\alpha_i(T+1)\beta_i(T+1)) \\
& \quad \textbf{\{definitions of } \alpha \textbf{ and } \beta \textbf{\}} \\
= \; & \sum_{i \in S} \left(\Pr\left(o_1^T, x_T = i \,\middle|\, \mu\right) \Pr\left(o_{T+1}^T \,\middle|\, \mu, x_T = i\right)\right) \\
& \quad \{o_{T+1}^T = \langle\rangle\} \\
= \; & \sum_{i \in S} \Pr\left(o_1^T, x_T = i \,\middle|\, \mu\right) \\
& \quad \textbf{\{math\}} \\
= \; & \Pr\left(o_1^T \,\middle|\, \mu\right)
\end{aligned}
$$

Again, this is exactly what we expect.

4 Best Path

The purpose of the Viterbi algorithm is to calculate the most likely state sequence given observations. However, we must also keep track of which parts of the observation sequence are attributed to each transition. This is not unlike the situation with HMMs which allow epsilon transitions: there you also need to keep track of which transitions emit observations, since transitions can be made which emit nothing.

We define a path as a sequence $P = \langle p_1 \ldots p_L \rangle$ such that p_i is a tuple $\langle t, x \rangle$ where t corresponds to the last of the (possibly multiple) observations made, and x refers to the state we were coming from when we output this observation (phrase).

Thus, we want to find:

$$\operatorname*{argmax}_P \Pr\left(P \,\middle|\, o_1^T, \mu\right) = \operatorname*{argmax}_P \Pr\left(P, o_1^T \,\middle|\, \mu\right)$$

Given a path $P = p_1^L$, we can calculate the states which correspond to the phrases of observations. Of course, we must constrain the paths such that their t component is monotonically increasing. If we write $p_i \langle t, x \rangle$ to mean that t is the t component of p_i and x is the x component, then we get that the observation phrase $o_{p_{i-1}.t+1}^{p_i.t}$ corresponds exactly to state $p_i \langle x, \_ \rangle$, where we implicitly define $p_0.t := 0$.

In order to calculate the P which maximizes the above expression, we introduce a dynamic programming algorithm which is an extension to the Viterbi algorithm for standard HMMs. To do this calculation, we wish to estimate the value:

$$\zeta_j(t) = \max_{l, p_1^{l-1}} \Pr\left(p_1^{l-1}, o_1^{t-1}, p_l.t = t-1, p_l.x = j \,\middle|\, \mu\right)$$

The idea here is that the path p_1^l accounts for observations o_1^{t-1} (and thus must end there); furthermore, the "next" element in the path, p_l should have j as its state. Thus $\zeta_j(t)$ is the probability of the most likely path which emits observations o_1^{t-1} and ends in state j. Given such a complete ζ table, we can calculate the probability of the optimal path as:

$$\begin{aligned}
\max_P \Pr\left(P, o_1^T \,\middle|\, \mu\right) &= \max_{l, p_1^{l-1}} \max_{j \in S} \Pr\left(p_1^{l-1}, o_1^T, p_l.t = T, p_l.x = j \,\middle|\, \mu\right) \\
&= \max_{j \in S} \zeta_j(T+1)
\end{aligned}$$

And, as usual, by storing in a secondary table ψ the tuple (t, x) we can calculate the optimal path by standard Viterbi methods.

We initialize the ζ table by setting:

$$\begin{aligned}
\zeta_j(1) &:= \max_{l, p_1^{l-1}} \Pr\left(p_1^{l-1}, o_1^{t-1}, p_l.t = t-1, p_l.x = j \,\middle|\, \mu\right) \\
&= \Pr\left(p_1^0, o_1^0, p_0.t = 0, p_1.x = j \,\middle|\, \mu\right) \\
&= \Pr\left(p_1.x = j \mid \mu\right) \\
&= \pi_j
\end{aligned}$$

This is essentially because there cannot be any path before outputting any observations, thus l must be 1, thus we are simply considering the first state in any path. We similarly define $\psi_j(1) := (0, \emptyset)$ to indicate that this is the start of the path.

For the inductive step $(t > 1)$, we define:

$$\begin{aligned}
\zeta_j(t) &:= \max_{l, p_1^{l-1}} \Pr\left(p_1^{l-1}, o_1^{t-1}, p_l.t = t-1, p_l.x = j \,\middle|\, \mu\right) \\
& \quad \textbf{\{Expand out } p_{l-2}\textbf{\}} \\
&= \max_{\substack{l, p_1^{l-2} \\ i \in S, t' < t}} \Pr\Big(p_1^{l-2}, o_1^{t'-1}, p_{l-1}.t = t'-1, p_{l-1}.x = i \\
& \qquad\qquad o_{t'}^{t-1}, p_l.t = t-1, p_l.x = j \,\Big|\, \mu\Big) \\
& \quad \textbf{\{Chain rule\}} \\
&= \max_{\substack{l, p_1^{l-2} \\ i \in S, t' < t}} \Pr\left(p_1^{l-2}, o_1^{t'-1}, p_{l-1}.t = t'-1, p_{l-1}.x = i \,\middle|\, \mu\right) \\
& \qquad\qquad \Pr\Big(o_{t'}^{t-1}, p_l.t = t-1, p_l.x = j \\
& \qquad\qquad\qquad \Big| \, p_1^{l-2}, o_1^{t'-1}, p_{l-1}.t = t'-1, p_{l-1}.x = i, \mu\Big) \\
& \quad \textbf{\{Definition of } \zeta\textbf{, and Marlov assumption\}} \\
&= \max_{\substack{l, p_1^{l-2} \\ i \in S, t' < t}} \zeta_i(t') \Pr\Big(o_{t'}^{t-1}, p_l.t = t-1, p_l.x = j \\
& \qquad\qquad\qquad \Big| \, p_{l-1}.x = i, \mu\Big) \\
& \quad \textbf{\{Definition of } a \textbf{ and } b\textbf{\}} \\
&= \max_{i \in S, t' < t} \zeta_i(t') a_{i,j} b_{i,j,o_{t'}^{t-1}}
\end{aligned}$$

And, of course, when we calculate $\zeta_j(t)$, we essentially need to choose an appropriate i and t', which we store in $\psi_j(t)$, so we can calculate the actual path at the end.

5 Parameter Re-estimation

The final problem we need to tackle in order to make the Phrase-Based HMM effective is that of parameter re-estimation. Essentially we want to find the model μ which best explains observations. There is no known analytic solution for standard HMMs, so we are fairly safe in assuming that it is far too difficult to find an analytic solution for this more complex problem. Thus, we also revert to an interative hill-climbing solution analogous to Baum-Welch re-estimation (i.e., the Forward Backward algorithm).

In order to achieve this, we define the re-esimated values $\hat{a}$ and $\hat{b}$ as:

$$\begin{aligned}
\hat{a}_{i,j} &= \frac{E\left[\# \text{ of transitions } i \rightsquigarrow j\right]}{E\left[\# \text{ of transitions } i \rightsquigarrow ?\right]} \\
\hat{b}_{i,j,\bar{k}} &= \frac{E\left[\# \text{ of transitions } i \rightsquigarrow j \text{ with } \bar{k} \text{ observed}\right]}{E\left[\# \text{ of transitions } i \rightsquigarrow j\right]}
\end{aligned}$$

In order to calculate these, we define:

$$\tau_{i,j}(t', t) = E\left[\# \text{ of transitions } i \rightsquigarrow j \text{ emitting } o_{t'}^{t}\right]$$

If we can calculate this, then we can re-estimate a and b as:

$$\begin{aligned}
\hat{a}_{i,j} &= \frac{\sum_{t'=1}^{T}\sum_{t=t'}^{T}\tau_{i,j}(t',t)}{\sum_{t'=1}^{T}\sum_{t=t'}^{T}\sum_{j'\in S}\tau_{i,j'}(t',t)} \\
\hat{b}_{i,j,\bar{k}} &= \frac{\sum_{t=1}^{T+1-|\bar{k}|}\delta(\bar{k}, o_t^{t+|\bar{k}|-1})\tau_{i,j}(t,t+|\bar{k}|-1)}{\sum_{t'=1}^{T}\sum_{t=t'}^{T}\tau_{i,j}(t',t)}
\end{aligned}$$

If we let $\eta_{i,j} = \sum_{t'=1}^{T}\sum_{t=t'}^{T}\tau_{i,j}(t',t)$, then we can rewrite these re-estimations as:

$$\begin{aligned}
\hat{a}_{i,j} &= \frac{\eta_{i,j}}{\sum_{j'\in S}\eta_{i,j'}} \\
\hat{b}_{i,j,\bar{k}} &= \frac{\sum_{t=1}^{T+1-|\bar{k}|}\delta(\bar{k}, o_t^{t+|\bar{k}|-1})\tau_{i,j}(t,t+|\bar{k}|-1)}{\eta_{i,j}}
\end{aligned}$$

So the parameter re-estimation problem boils down to calculating the τ table. We can do this using a combination of the forward and backward probabilities. Recall that these are defined as:

$$\begin{aligned}
\alpha_j(t) &= \Pr\left(o_1^{t-1}, x_{t-1} = j \,\middle|\, \mu\right) \\
\beta_i(t) &= \Pr\left(o_t^T \,\middle|\, \mu, x_{t-1} = i\right)
\end{aligned}$$

Now, we are set to calculate τ:

$$\begin{aligned}
& \tau_{i,j}(t',t) \\
& \textbf{\{definition of } \tau \textbf{\}} \\
= \; & E\left[\#\text{ of transitions } i \rightsquigarrow j \text{ emitting } o_{t'}^{t}\right] \\
& \textbf{\{calculating the probability explicitly\}} \\
= \; & \Pr\left(x_{t'-1} = i, x_t = j \,\middle|\, o_1^T, \mu\right) \\
& \textbf{\{Bayes' Rule\}} \\
= \; & \frac{\Pr\left(x_{t'-1} = i, x_t = j, o_1^T \,\middle|\, \mu\right)}{\Pr\left(o_1^T \,\middle|\, \mu\right)} \\
& \textbf{\{split the observation sequence in half\}} \\
= \; & \frac{\Pr\left(x_{t'-1} = i, o_1^{t'-1}, o_{t'}^T, x_t = j \,\middle|\, \mu\right)}{\Pr\left(o_1^T \,\middle|\, \mu\right)} \\
& \textbf{\{reverse chain rule\}} \\
= \; & \frac{\Pr\left(x_{t'-1} = i, o_1^{t'-1} \,\middle|\, \mu\right)\Pr\left(o_{t'}^T, x_t = j \,\middle|\, x_{t'-1} = i, o_1^{t'-1}, \mu\right)}{\Pr\left(o_1^T \,\middle|\, \mu\right)} \\
& \textbf{\{definition of } \alpha \textbf{ and reverse chain rule\}}
\end{aligned}$$

$$= \frac{\alpha_i(t') \Pr\left(x_t = j \,\middle|\, x_{t'-1} = i, o_1^{t'-1}, \mu\right) \Pr\left(o_{t'}^T \,\middle|\, x_{t'-1} = i, o_1^{t'-1}, x_t = j, \mu\right)}{\Pr\left(o_1^T \,\middle|\, \mu\right)}$$

{splitting the observation sequence again and reverse chain rule}

$$= \frac{\alpha_i(t') \Pr\left(x_t = j \,\middle|\, x_{t'-1} = i, o_1^{t'-1}, \mu\right) \Pr\left(o_{t'}^t, o_{t+1}^T \,\middle|\, x_{t'-1} = i, o_1^{t'-1}, x_t = j, \mu\right)}{\Pr\left(o_1^T \,\middle|\, \mu\right)}$$

{definition of a and b, and Markov assumption}

$$= \frac{\alpha_i(t') a_{i,j} b_{i,j,o_{t'}^t} \Pr\left(o_{t+1}^T \,\middle|\, x_t = j, \mu\right)}{\Pr\left(o_1^T \,\middle|\, \mu\right)}$$

{definition of β}

$$= \frac{\alpha_i(t') a_{i,j} b_{i,j,o_{t'}^t} \beta_j(t+1)}{\Pr\left(o_1^T \,\middle|\, \mu\right)}$$

6 Summary and Complexity Analysis

Recall the final definitions from the previous sections:

$$\begin{aligned}
\alpha_j(t+1) &= \textstyle\sum_{t'=0}^{t-1} \sum_{i \in S} \left(\alpha_i(t'+1) \cdot a_{i,j} \cdot b_{i,j,o_{t'+1}^t}\right) && , \forall j \in S, 1 \le t \le T \\
\beta_i(t) &= \textstyle\sum_{t'=t}^{T} \sum_{j \in S} \left(a_{i,j} \cdot b_{i,j,o_t^{t'}} \cdot \beta_j(t'+1)\right) && , \forall i \in S, 1 \le t \le T \\
\tau_{i,j}(t', t) &= \frac{\alpha_i(t') a_{i,j} b_{i,j,o_{t'}^t} \beta_j(t+1)}{\Pr\left(o_1^T \,\middle|\, \mu\right)} && , \forall i, j \in S, 1 \le t' \le t \le T \\
\eta_{i,j} &= \textstyle\sum_{t'=1}^{T} \sum_{t=t'}^{T} \tau_{i,j}(t', t) && , \forall i, j \in S \\
\hat{a}_{i,j} &= \frac{\eta_{i,j}}{\sum_{j' \in S} \eta_{i,j'}} && , \forall i, j \in S \\
\hat{b}_{i,j,\bar{k}} &= \frac{\sum_{t=1}^{T+1-|\bar{k}|} \delta(\bar{k}, o_t^{t+|\bar{k}|-1}) \tau_{i,j}(t, t+|\bar{k}|-1)}{\eta_{i,j}} && , \forall i, j \in S, \bar{k} \in K^+
\end{aligned}$$

Of course, we needn't consider all $\bar{k} \in K^+$ for the re-estimation of b: only the ones which actually appear in o_1^T.

Calculating the α, β and τ tables will take time $\mathcal{O}\left(N^2 T^2\right)$ each. Calculating the η table will only take time $\mathcal{O}\left(N^2\right)$. Re-estimating $\hat{a}$ will take time $\mathcal{O}\left(N^2\right)$ assuming we are decently intelligent about calculating the denominator. Finally, re-estimating $\hat{b}$ will take time $\mathcal{O}\left(N^2 T^2\right)$ since we need to consider every subsequence of o_1^T. Thus, all in all, the re-estimation will take $\mathcal{O}\left(N^2 T^2\right)$ time.

In many applications we needn't actually consider all subsequences of observations. Usually we will want to bound the length of the sequence; otherwise we are unlikely to encounter enough training data to get reasonable estimates of emission probabilities. If we enforce a maximum observation sequence length of l, then all the factors of T^2 drop to Tl, a fairly substantial improvement, leaving us with a complexity of $\mathcal{O}\left(N^2 T l\right)$ for parameter re-estimation (typically l will be much smaller than T).

Moreover, if the HMM is sparse, then the sums over all $i, j \in S$ need only be a sum over all the $i \in S$ and j such that there is an edge from i to j. If the maximum out-degree of any node is b, all the N^2 factors drop to Nb, leaving us with a final complexity of $\mathcal{O}\left(NTbl\right)$.