Probabilistic finite-state grammars

Probabilistic finite-state automata (PFSAs) are also known as *hidden markov models*. A markov model in general is anything with the core generative structure of the probabilistic bigram grammars we saw last week. Probabilistic FSAs are *hidden* markov models because of the fact that the markovian transitions are transitions between underlying states rather than transitions between the visible surface symbols. This "hiddenness" is the same property that allows an FSA to "generate the same word-sequence two ways", and therefore provide at least a very simple model of what we might call structural ambiguity.

1 Summing probabilities across alternatives

When a certain event can happen in two (or more) different ways, we often end up needing to calculate the probability of this event by summing the probabilities of the two (or more) possibilities. This comes up a lot with PFSAs. Here are a couple of examples to introduce the idea.

1.1 A simple non-linguistic example

Imagine a simple game played with a bag containing 4 red balls and 5 yellow balls, a (fair) six-sided die and a (fair) coin. You begin by picking a ball from the bag, and then:

- If the ball you choose is red, then you flip the coin; if the coin comes up heads, then you win (and otherwise, you lose).
- If the ball you choose is yellow, then you roll the die; if the die comes up six, then you win (and otherwise, you lose).

What is the probability of winning the game?

$$\begin{aligned}
\Pr(\text{win}) &= \Pr(B = \text{red}) \Pr(\text{win} \mid B = \text{red}) + \Pr(B = \text{yellow}) \Pr(\text{win} \mid B = \text{yellow}) && (1)\\
&= \Pr(B = \text{red}) \Pr(C = \text{heads}) + \Pr(B = \text{yellow}) \Pr(D = \text{six}) && (2)\\
&= (\tfrac{4}{9} \times \tfrac{1}{2}) + (\tfrac{5}{9} \times \tfrac{1}{6}) && (3)
\end{aligned}$$

1.2 Summing in bigram grammars

Suppose we have a probabilistic bigram grammar over the symbols {a,b}, defined by the following transition probabilities:

$$\begin{array}{ccc}
\Pr(W_i = \text{a} \mid W_{i-1} = \texttt{<s>}) = 0.6 & \Pr(W_i = \text{a} \mid W_{i-1} = \text{a}) = 0.3 & \Pr(W_i = \text{a} \mid W_{i-1} = \text{b}) = 0.5 \\
\Pr(W_i = \text{b} \mid W_{i-1} = \texttt{<s>}) = 0.4 & \Pr(W_i = \text{b} \mid W_{i-1} = \text{a}) = 0.6 & \Pr(W_i = \text{b} \mid W_{i-1} = \text{b}) = 0.3 \\
\Pr(W_i = \texttt{</s>} \mid W_{i-1} = \texttt{<s>}) = 0 & \Pr(W_i = \texttt{</s>} \mid W_{i-1} = \text{a}) = 0.1 & \Pr(W_i = \texttt{</s>} \mid W_{i-1} = \text{b}) = 0.2
\end{array}$$

One question we might ask now is: What's the probability of generating some symbol-sequence which has 'a' as the second symbol? Notice that we can't just read this directly from the transition probabilities above (in the way that we can immediately read off the fact that there's a 0.6 probability of generating 'a' as the first symbol), because we have to deal with an "unknown" first symbol. But we can compute the probability we're interested in by summing over the two possible outcomes for that first symbol:

$$\begin{aligned}
\Pr(W_2 = \text{a}) &= \Pr(W_1 = \text{a}, W_2 = \text{a}) + \Pr(W_1 = \text{b}, W_2 = \text{a}) && (4)\\
&= \Pr(W_1 = \text{a}) \Pr(W_2 = \text{a} \mid W_1 = \text{a}) + \Pr(W_1 = \text{b}) \Pr(W_2 = \text{a} \mid W_1 = \text{b}) && (5)\\
&= 0.6 \times 0.3 + 0.4 \times 0.5 && (6)
\end{aligned}$$

Suppose now, just for the sake of illustration, that we knew the probability of 'a' appearing as the tenth symbol generated and the probability of 'b' appearing as the tenth symbol generated, i.e. we are given specific numbers for $\Pr(W_{10} = \text{a})$ and $\Pr(W_{10} = \text{b})$. How can we work out the probability of 'a' appearing as the eleventh symbol? This is very simple, we just follow the same logic:

$$\begin{aligned}
\Pr(W_{11} = \text{a}) &= \Pr(W_{10} = \text{a}, W_{11} = \text{a}) + \Pr(W_{10} = \text{b}, W_{11} = \text{a}) && (7)\\
&= \Pr(W_{10} = \text{a}) \Pr(W_{11} = \text{a} \mid W_{10} = \text{a}) + \Pr(W_{10} = \text{b}) \Pr(W_{11} = \text{a} \mid W_{10} = \text{b}) && (8)\\
&= \Pr(W_{10} = \text{a}) \times 0.3 + \Pr(W_{10} = \text{b}) \times 0.5 && (9)
\end{aligned}$$

Notice how when we write, for example, $\Pr(W_{10} = \text{b}, W_{11} = \text{a})$ we've combined (or implicitly summed over) all the different options for W_9, W_8, and so on back down to W_1. We can do this because all we really need to think about are the two *equivalence classes* of outcomes in all of these earlier positions: those where $W_{10} = \text{a}$, and those where $W_{10} = \text{b}$.

1.3 The general pattern

If we're interested in the probability of some event A, and there are multiple "routes to" A which we can call B_1, $B_2 \dots B_m$ (such that an occurrence of A must co-occur with exactly one of these B_1, B_2, $\dots B_m$), then

$$\begin{aligned}
\Pr(A) &= \sum_{X \in \{B_1, B_2, \dots, B_m\}} \Pr(A, X) &&= \Pr(A, B_1) + \dots + \Pr(A, B_m)\\
&= \sum_{X \in \{B_1, B_2, \dots, B_m\}} \Pr(A \mid X) \Pr(X) &&= \Pr(A \mid B_1) \Pr(B_1) + \dots + \Pr(A \mid B_m) \Pr(B_m)
\end{aligned}$$

As an abstract preview of how this will be relevant: think of B_1, B_2, $\dots B_m$ as (the hidden parts of) structural descriptions that might constitute a way of generating some surface form (say, a list of strings). Notice that, if we're interested in some surface word-sequence $w_1 w_2 \dots w_n$, our grammars have had the general property that

$$w_1 w_2 \dots w_n \text{ is generated} \quad \text{iff} \quad \exists sd\, [\texttt{pf}\ sd = w_1 w_2 \dots w_n \ \wedge\ \texttt{wellFormed}\ sd]$$

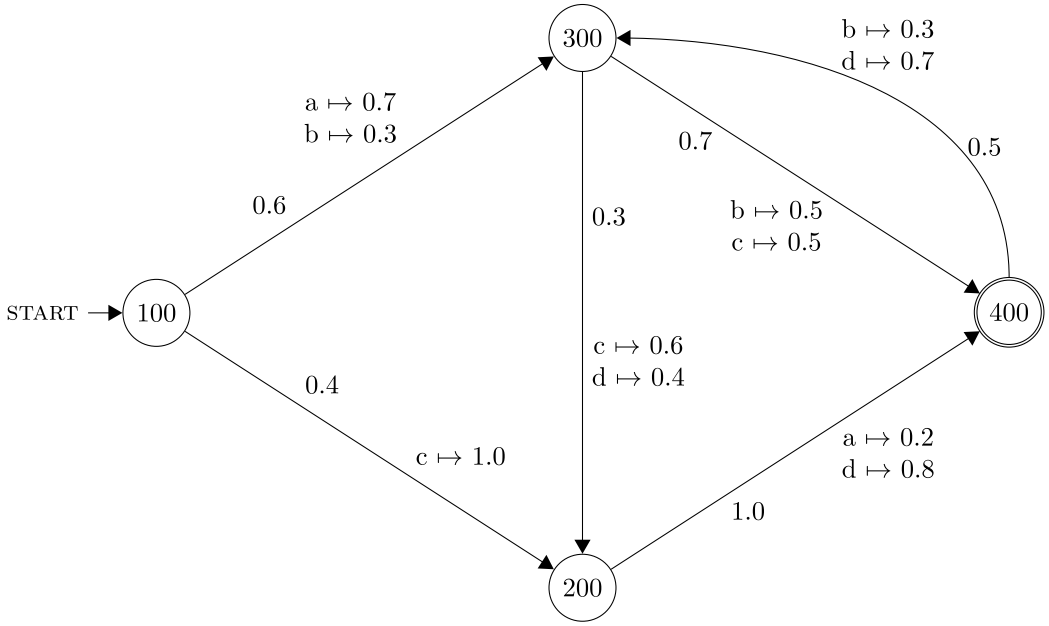
which might be re-expressed as

$$w_1 w_2 \dots w_n \text{ is generated} \quad \text{iff} \quad \bigvee_{sd} [\texttt{pf}\ sd = w_1 w_2 \dots w_n \ \wedge\ \texttt{wellFormed}\ sd]$$

If you think of generating $w_1 w_2 \dots w_n$ as the event A from above, then choosing a structural description sd corresponds to choosing an X; and `wellFormed` sd plays a role roughly analogous to $\Pr(X)$, and `pf` $sd = w_1 w_2 \dots w_n$ plays a role roughly analogous to $\Pr(A \mid X)$.

2 Probabilistic finite-state grammars

Here's a first example of a probabilistic finite-state grammar.



There are two kinds of probability distributions here.

- For each state, there is a probability distribution over the outward arcs. One probability of this sort of shown on each arc in the diagram. These are known as **transition probabilities**. Some examples:

$$\begin{aligned}\Pr(S_i = 300 \mid S_{i-1} = 100) = 0.6\\ \Pr(S_i = 200 \mid S_{i-1} = 100) = 0.4\\ \Pr(S_i = 400 \mid S_{i-1} = 200) = 1.0\end{aligned}$$

  In addition, the 'START' marker tells us that $\Pr(S_0 = 100) = 1.0$; and since the probabilities of the outgoing arcs from state 400 only add up to 0.5, we know there is a 0.5 probability of ending in that state.

- For each arc, there is a probability distribution over output symbols. These are known as **emission probabilities**. Some examples:

$$\begin{aligned}\Pr(W_i = \text{a} \mid S_i = 100, S_{i+1} = 300) = 0.7\\ \Pr(W_i = \text{b} \mid S_i = 100, S_{i+1} = 300) = 0.3\\ \Pr(W_i = \text{a} \mid S_i = 200, S_{i+1} = 400) = 0.2\\ \Pr(W_i = \text{d} \mid S_i = 200, S_{i+1} = 400) = 0.8\end{aligned}$$

  Note that we are using W_i to describe the word that is emitted upon the transition from S_i to S_{i+1}, i.e. the index of the word matches the index of the "from state", not the index of the "to state".

Some example calculations:

- What is the probability of starting in state 100, then emitting 'a' as we move to state 300, and then emitting 'c' as we move to state 200 (whatever may happen after that)?

$$\begin{aligned}
&\Pr(S_0 = 100, W_0 = \text{a}, S_1 = 300, W_1 = \text{c}, S_2 = 200) \\
=\ &\Pr(S_0 = 100) \times \big(\Pr(S_1 = 300 \mid S_0 = 100) \times \Pr(W_0 = \text{a} \mid S_0 = 100, S_1 = 300)\big) \times \\
&\quad \big(\Pr(S_2 = 200 \mid S_1 = 300) \times \Pr(W_0 = \text{c} \mid S_1 = 300, S_2 = 200)\big) \\
=\ &1.0 \times (0.6 \times 0.7) \times (0.3 \times 0.6)
\end{aligned}$$

- What is the probability of (starting, then) emitting 'a', then emitting 'c' (whatever may happen after that)?

$$\begin{aligned}
&\Pr(W_0 = \text{a}, W_1 = \text{c}) \\
=\ &\sum_{x,y,z \in \{100,200,300,400\}} \big[\Pr(S_0 = x) \times \big(\Pr(S_1 = y \mid S_0 = x) \times \Pr(W_0 = \text{a} \mid S_0 = x, S_1 = y)\big) \times \\
&\qquad \big(\Pr(S_2 = z \mid S_1 = y) \times \Pr(W_0 = \text{c} \mid S_1 = y, S_2 = z)\big)\big] \\
=\ &\Pr(S_0 = 100) \times \big(\Pr(S_1 = 300 \mid S_0 = 100) \times \Pr(W_0 = \text{a} \mid S_0 = 100, S_1 = 300)\big) \times \\
&\quad \big(\Pr(S_2 = 200 \mid S_1 = 300) \times \Pr(W_0 = \text{c} \mid S_1 = 300, S_2 = 200)\big) \\
&+ \\
&\Pr(S_0 = 100) \times \big(\Pr(S_1 = 300 \mid S_0 = 100) \times \Pr(W_0 = \text{a} \mid S_0 = 100, S_1 = 300)\big) \times \\
&\quad \big(\Pr(S_2 = 400 \mid S_1 = 300) \times \Pr(W_0 = \text{c} \mid S_1 = 300, S_2 = 400)\big) \\
=\ &1.0 \times (0.6 \times 0.7) \times (0.3 \times 0.6) + 1.0 \times (0.6 \times 0.7) \times (0.7 \times 0.5)
\end{aligned}$$

- Suppose we knew the distibution of the possible states we could reach after eight steps, i.e. we were given concrete numbers for $\Pr(S_8 = 100)$, $\Pr(S_8 = 200)$, $\Pr(S_8 = 300)$ and $\Pr(S_8 = 400)$. How would we calculate the probability $\Pr(W_8 = \text{c})$ from these?

$$\begin{aligned}
&\Pr(W_8 = \text{c}) \\
=\ &\sum_{x,y \in \{100,200,300,400\}} \big[\Pr(S_8 = x) \times \Pr(S_9 = y \mid S_8 = x) \times \Pr(W_9 = \text{c} \mid S_8 = x, S_9 = y)\big] \\
=\ &\Pr(S_8 = 100) \times \Pr(S_9 = 200 \mid S_8 = 100) \times \Pr(W_9 = \text{c} \mid S_8 = 100, S_9 = 200) \\
&+ \Pr(S_8 = 300) \times \Pr(S_9 = 200 \mid S_8 = 300) \times \Pr(W_9 = \text{c} \mid S_8 = 300, S_9 = 200) \\
&+ \Pr(S_8 = 300) \times \Pr(S_9 = 400 \mid S_8 = 300) \times \Pr(W_9 = \text{c} \mid S_8 = 300, S_9 = 400) \\
=\ &\Pr(S_8 = 100) \times 0.4 \times 1.0 \\
&+ \Pr(S_8 = 300) \times 0.3 \times 0.6 \\
&+ \Pr(S_8 = 300) \times 0.7 \times 0.5
\end{aligned}$$

3 Forward and backward probabilities

There are (famously) three kinds of questions we might want to ask about these grammars:

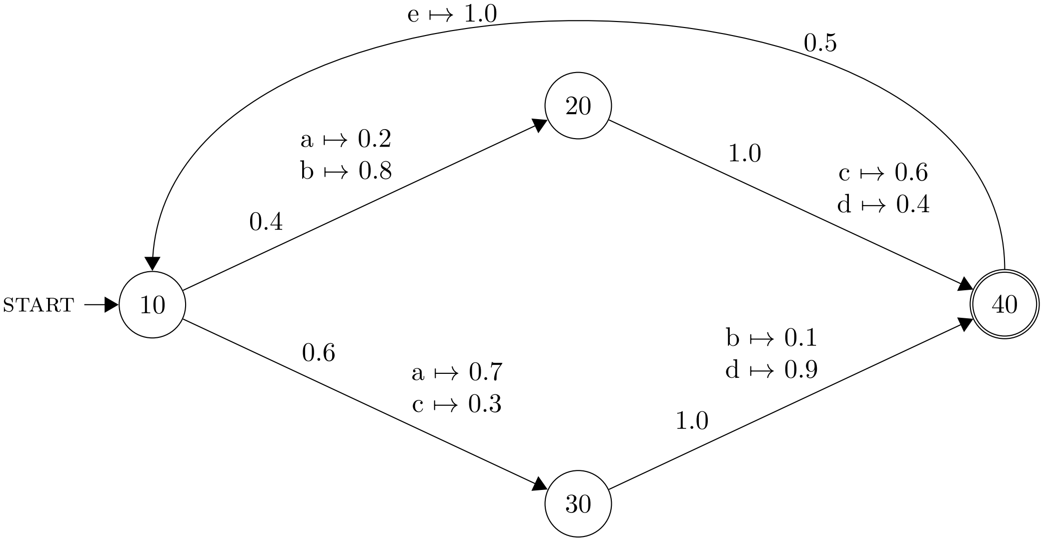
(10) a. Given a PFSA, what is the probability of a particular symbol-sequence $w_1 w_2 \dots w_n$ being generated?

b. Given a PFSA and a particular symbol-sequence $w_1 w_2 \dots w_n$, what state sequence best explains an observation of $w_1 w_2 \dots w_n$?

c. Given some symbol-sequence $w_1 w_2 \dots w_n$ a plain FSA, what values for the transition and emission probabilities best explains an observation $w_1 w_2 \dots w_n$?

Calculation of what are known as *forward probabilities* and *backward probabilities* plays a major role in the first and third of these questions, and the same kind of logic gets you most of the way towards answering the second question.

Let's switch to a slightly more interesting example grammar.



And let's suppose that we have Haskell functions that allow us to query this grammar as follows:

```
trProb :: State -> State -> Probability
emProb :: (State,State) -> Symbol -> Probability
startProb :: State -> Probability
endProb :: State -> Probability
allStates :: [State]
```

3.1 Forward probabilities

Notice there are two ways in which this grammar can, from its start state, generate 'a d' and get to state 40. The probability of this happening (in either of those two ways) is known as a *forward probability*.

More generally: we'd like to write a function `probForward :: [Symbol] -> State -> Probability` such that `probForward` $w_0 \dots w_n$ st is the probability of generating $w_0 \dots w_n$ as the first n symbols and also being in state st after taking those first n transitions.

Put differently:

$$\texttt{probForward}\ w_0 \dots w_n\ st = \Pr(W_0 = w_0, W_1 = w_1, \dots, W_n = w_n, S_{n+1} = st) \tag{11}$$

This can be re-expressed as a sum over all the possible sequences of earlier unknown states $s_0, s_1, \dots, s_n$. But the crucial insight for calculating forward probabilities is to notice that they can be expressed recursively, like this, where we sum over one previous state at a time:

$$\begin{aligned} &\texttt{probForward}\ w_0 \dots w_{n-1} w_n\ st \\ &\quad = \sum_{prev \in \texttt{allStates}} \big[(\texttt{probForward}\ w_0 \dots w_{n-1}\ prev) \times (\texttt{trProb}\ prev\ st) \times (\texttt{emProb}\ (prev, st)\ w_n)\big] \end{aligned} \tag{12}$$

Of course this will only work if we can also provide values for the base case. But this is easy (once you think about it for a moment): the probability of the best sequence emitting no output ending at state st is just the starting probability of st.

$$\texttt{probForward}\ [\,]\ st \quad = \quad \texttt{startProb}\ st \tag{13}$$

3.2 Backward probabilities

We can also flip things around — to switch to thinking of the *ending parts* of a string as the string's subconstituents, with states representing equivalence classes of such subconstituents, in accord with the way we have generalized up to trees — and work with *backward probabilities* instead. As we've seen, the probability of starting and then generating 'a d' and getting to state 40 is a forward probability; the probability, given that we start in state 10, of generating 'a d' and then ending is a backward probability.

More generally: we'd like to write a function `probBackward :: [Symbol] -> State -> Probability` such that `probBackward` $w_1 \dots w_n$ st is the probability, given that we start in state st, of generating $w_1 \dots w_n$ and then ending.

Put differently:

$$\texttt{probBackward}\ w_0 \dots w_n\ st = \Pr(W_k = w_0, W_{k+1} = w_1, \dots, W_{k+n} = w_n, \text{end on } S_{k+n+1} \mid S_k = st) \tag{14}$$

This can also be (wait for it) expressed recursively:

$$\begin{aligned} &\texttt{probBackward}\ w_1 w_2 \dots w_n\ st \\ &\quad = \sum_{next \in \texttt{allStates}} \big[(\texttt{trProb}\ st\ next) \times (\texttt{emProb}\ (st, next)\ w_1) \times (\texttt{probBackward}\ w_2 \dots w_n\ next)\big] \end{aligned} \tag{15}$$

$$\texttt{probBackward}\ [\,]\ st \quad = \quad \texttt{endProb}\ st \tag{16}$$

3.3 Combining forward and backward probabilities

Notice that forward probabilities take into account starting probabilities, but must be multiplied by ending probabilities to get "complete" probabilities; backward probabilities take into account ending probabilities, but must be multiplied by starting probabilities to get "complete" probabilities. So to find the probability of generating, say, 'a d' as our complete output in any way at all, we can either (i) use forward probabilities and sum over possible end states, or (ii) use backward probabilities and sum over possible start states.

$$\sum_{st \in \texttt{allStates}} \big[(\texttt{probForward ["a","d"]}\ st) \times (\texttt{endProb}\ st)\big] \tag{17}$$

$$\sum_{st \in \texttt{allStates}} \Big[(\texttt{startProb}\ st) \times (\texttt{probBackward ["a","d"]}\ st) \Big] \tag{18}$$

But these are actually just special cases of a more general formula, which neatly expresses the familiar idea about equivalent subexpressions being interchangeable:

$$\Pr(w_1 \dots w_n) = \sum_{st \in \texttt{allStates}} \Big[(\texttt{probForward}\ w_1 \dots w_i\ st) \times (\texttt{probBackward}\ w_{i+1} \dots w_n\ st) \Big] \tag{19}$$

The logic of this equation is the basis of the *forward-backward algorithm*, which is used to find the transition and emission probabilities that best explain some observed symbol-sequence (i.e. question (10c)).