CS 4850: Mathematical Foundations for the Information Age

Lecture #39 - April 24, 2009

Scribes: Scott Rogoff and Andrew Owens

Hidden Markov Models

1. How probable is an output sequence $O_0, O_1, \ldots, O_T$?

For each $t \leq T$ calculate the probability of the sequence $O_0, O_1, \ldots, O_t$ and ending at state i.

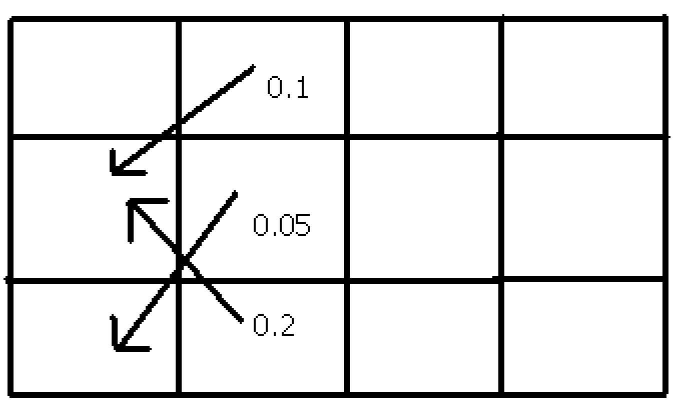
t = 0: $\pi(i)P(O_0|i)$

t + 1: $\sum_j P(O_0, O_1, \ldots, O_t \textit{ and ending in state } j)P(O_{t+1}|\, j)\, a_{ij}$

This algorithm is $O(Tn^2)$ and uses $O(n)$ space.

2. Given an output sequence, what is the most likely sequence of states (Viterbi algorithm)?

Maintain probability for most likely sequence of states. Create a table where entry (j, t) is the probability of most likely path that ended in state j at time t. Also keep track of the previous state in the most likely path:



3. Compute a model given an upper bound on the number of states, n, and a sequence $O_0, O_1, \ldots, O_T$ where T >> n.

Definitions:

$\alpha_t(i)$ is the probability of seeing the $O_0, O_1, \ldots, O_t$ and ending at state i at time t.

$\beta_{t+1}(j)$ is the probability of seeing the remainder of the sequence giving that you are in state j at time t+1.

$\delta_t(i,j)$ is the probability of going from state i to j at time t given output O.

$\gamma_t(i)$ is the probability of being in state i at time t given O.

$P(O)$ is the probability of the output sequence.

Calculations:

$$\delta_t(i,j) = \frac{\alpha_t(i) a_{ij} b_j(O_{t+1}) \beta_{t+1}(j)}{P(O)} = \frac{P(given\ seq\ O, go\ to\ state\ i, transition\ to\ j, then\ go\ on)}{P(O)}$$

$$P(O) = \sum_{i,j} \alpha_t(i) a_{ij} b_j(O_{t+1}) \beta_{t+1}(j)$$

$$\gamma_t(i) = \sum_j \delta_t(i,j)$$

Estimate a_{ij} and $b_j(o_k)$:

$$a_{ij} = \frac{\sum_{t=1}^{T-1} \delta_t(i,j)}{\sum_{t=1}^{T-1} \gamma_t(i)}$$

$$b_j(o_k) = \frac{\sum_{t\ and\ O_t=o_k} \gamma_t(j)}{\sum_t \gamma_t(j)}$$

<u>Ranking</u>

Here we consider the following problem: given a set of items to be ranked, and a complete ranking of these items from n different sources, how can we combine these into one global ranking?

Example: Assume there are three papers presented at a conference called a, b, and c, and a panel of three judges are to rank them from most to least important. Here's an example of such a ranking:

Individual	First paper	Second paper	Third paper
1	a	b	c
2	b	c	a
3	c	a	b

To try to determine which paper is the winner, we might ask how often certain papers were ranked above others. Here, two of three people ranked a above b. Thus we might argue that a should be ranked above b in the global ranking. We will denote this by a > b. However, by the same argument we could say that b > c and c > a. There is a breakdown of transitivity here, because the first two statements would imply that c < a.

In order to avoid ranking schemes that seem to have intuitive flaws like this, we will try to construct a set of axioms that any ranking system should have.

Axioms:

- *non-dictatorship*: the algorithm doesn't just choose one individual's ranking and output it as the global ranking
- *unanimity*: if every individual ranks a above b, then the global ranking must prefer a to b
- *independence of irrelevant alternatives*: if individuals modify their ranks but leave the order of a and b the same, the global order of a and b shouldn't change

We would hope to be able to find an algorithm that obeys all three of these axioms. Unfortunately, the following theorem shows this is impossible.

Theorem (Arrow) – Any algorithm for creating a global ranking of ≥ 3 objects that satisfies unanimity and independence of irrelevant alternatives is a dictatorship.

The proof of this theorem is given in the next lecture.