COMS 6998, Spring 2012: Problem Set 2

Total points: 75 points

Due date: 5pm, 9th April 2012

Late policy: 5 points off for every day late, 0 points if handed in after 5pm on 13th April

Question 1 (15 points)

In this question we'll derive an EM approach to word clustering. In this model, the training sample $x^1, x^2, \ldots x^m$ is a sequence of m bigrams of the following form: each x^i is of the form w_1^i, w_2^i where w_1^i, w_2^i are words, and w_2^i is seen following w_1^i in the corpus. The hidden variables y can take one of K values, $1, 2, \ldots K$. The model is defined as follows:

$$p(w_2, y|w_1; \Theta) = q(y|w_1)r(w_2|y)$$

Thus if $\mathcal{V}$ is the vocabulary—the set of possible words in any document—the parameters in the model are:

- $q(y|w)$ for $y = 1 \ldots K$, for $w \in \mathcal{V}$
- $r(w|y)$ for $y = 1 \ldots K$ and $w \in \mathcal{V}$

Our aim in this question will be to derive EM updates which optimize the log-likelihood of the data:

$$L(\Theta) = \sum_{i=1}^{m} \log p(w_2^i|w_1^i; \Theta) = \sum_{i=1}^{m} \log \sum_y q(y|w_1^i)r(w_2^i|y)$$

Give pseudo-code showing how to derive an updated parameter vector Θ^t from a previous parameter vector Θ^{t-1}. I.e., show pseudo-code that takes as input parameter estimates $q^{t-1}(y|w)$ for all y, w and $r^{t-1}(w|y)$ for all w, y, and as output provides updated parameter estimates $q^t(y|w)$ and $r^t(w|y)$ using EM.

Question 2 (30 points)

In lecture (see also the accompanying note on EM) we saw how the forward-backward algorithm could be used to efficiently calculate probabilities of the following form for an HMM:

$$P(y_j = p|x; \Theta) = \sum_{y:y_j=p} p(y|x; \Theta)$$

and

$$P(y_j = p, y_{j+1} = q|x; \Theta) = \sum_{y:y_j=p,y_{j+1}=q} p(y|x, \Theta)$$

where x is some sequence of output symbols, and Θ are the parameters of the model. Here y_j is the j'th state in a state sequence y, and p, q are integers in the range $1 \ldots N$ assuming an N state HMM.

Question 2(a) (10 points) State how the following quantity can be calculated in terms of the forward-backward probabilities, and some of the parameters in the model:

$$P(y_2 = 1, y_3 = 2, y_4 = 1|x; \Theta)$$

(we assume that the sequence x is of length at least 4)

Question 2(b) (10 points) State how the following quantity can be calculated in terms of the forward-backward probabilities, and some of the parameters in the model:

$$P(y_2 = 1, y_5 = 1|x; \Theta)$$

(we assume that the sequence x is of length at least 5. Don't worry too much about the efficiency of your solution: we **do** expect you to use forward and backward terms, but we **don't** expect you to calculate any other quantities using dynamic programming.)

Question 2(c) (10 points) Say that we now wanted to calculate probabilities for an HMM such as the following:

$$\max_{y:y_j=p} p(y|x; \Theta)$$

so this is the maximum probability of any state sequence underlying x, with the constraint that the j'th label y_j is equal to p.

How would you modify the definition of the forward and backward terms—i.e., the recursive method for calculating them—to support this kind of calculation? How would you then calculate

$$\max_{y:y_3=1} p(y|x; \Theta)$$

assuming that the input sequence x is of length at least 3?

Question 3 (30 points)

We are now going to derive an EM-style model for machine translation. Each training example in our data consists of a French sentence f together with an English sentence e. Each French sentence f consists of m words $f_1 \dots f_m$; each English sentence e consists of l words $e_1 \dots e_l$ (for simplicity we assume that French sentences always have length m, and English sentences always have length l).

Our goal is to build a model of

$$p(f|e)$$

i.e., the conditional probability of a French sentence f given an English sentence e. To do this we will introduce hidden variables $a_1 \dots a_m$. Each hidden variable a_j specifies which English word f_j is "aligned" to. Each a_j can take any value in $1, 2, \dots l$.

For some background on this type of model, the note at

`http://www.cs.columbia.edu/~mcollins/courses/nlp2011/notes/ibm12.pdf`

may be useful.

The model then takes the following form:

$$p(f, a|e) = \prod_{j=1}^{m} d(a_j|a_{j-1}) \prod_{j=1}^{m} t(f_j|e_{a_j})$$

Here we define a_0 to be `NULL`, where `NULL` is a special state in the model. $p(f, a|e)$ is the joint probability of alignment sequence $a = a_1 \dots a_m$, together with French words $f = f_1 \dots f_m$, conditioned on English sentence $e = e_1 \dots e_l$.

The parameters in this model are as follows:

- $d(a'|a)$ is the conditional probability of seeing alignment value a', given that the previous alignment value was a.
- $t(f|e)$ for any French word f and English word e is the conditional probability of seeing French word f, given that English word e is being translated.

Note that this model form is quite similar to a bigram HMM model. The d parameters are very similar to transition parameters in a regular HMM. The t parameters are very similar to emission parameters in an HMM. The only real difference is that the emission probabilities vary depending on the English sentence being translated.

Question 3(a) (10 points)

Assume that we have parameter values $d(a'|a)$ and $t(f|e)$. State how the forward-backward algorithm can be used to calculate

$$P(a_j = k|f, e) = \sum_{a:a_j=k} p(a|f, e)$$

for any $j \in \{1 \dots m\}$, $k \in \{1 \dots l\}$, for a fixed French sentence f and English sentence e.

Question 3(b) (10 points)

Assume that we have parameter values $d(a'|a)$ and $t(f|e)$. State how the forward-backward algorithm can be used to calculate

$$P(a_j = k, a_{j+1} = k'|f, e) = \sum_{a:a_j=k,a_{j+1}=k'} p(a|f, e)$$

for any $j \in \{1 \dots m-1\}$, $k \in \{1 \dots l\}$, $k' \in \{1 \dots l\}$, for a fixed French sentence f and English sentence e.

Question 3(c) (10 points)

Now assume that we have training examples $f^{(i)}, e^{(i)}$ for $i = 1 \dots n$, where each $f^{(i)}$ is a French sentence, and each $e^{(i)}$ is an English sentence. Give pseudo-code for an EM algorithm that trains the $d(a'|a)$ and $t(f|e)$ parameters of the model. (Hint: the quantities computed in questions 3(b) and 3(c) should be very useful; the algorithm should be very similar to the EM algorithm for HMMs given in class.)