Assignment: applying brute force EM on IBM model 1

November 16, 2017

There were some notes showing a worked example of brute force EM training with IBM model 1. This assignment varies the example a bit.

Suppose a corpus of 3 pairs:

<table>
<thead>
<tr>
<th>$s^1$</th>
<th>green house</th>
<th>$s^2$</th>
<th>the house</th>
</tr>
</thead>
<tbody>
<tr>
<td>$o^1$</td>
<td>casa verde</td>
<td>$o^2$</td>
<td>la casa</td>
</tr>
</tbody>
</table>

Below the 1st iteration of the brute force EM algorithm is shown. You need to complete the 2nd and 3rd iteration.

To apply the brute force EM algorithm, for each pair, each of its possible alignments has to be considered. In the real IBM Model 1, a possibility is to align positions in $o$ with NULL, and in the worked example we excluded that. In this assignment we are going to restrict the possibilities even further: the alignments will be 1-to-1, and not possibly n-to-1: so if $i$ is the value $a(j)$ for some $j$ it cannot be the value for any other $j'$. This means there are just 2 possibilities for each pair:

- use $a^1_1, a^1_2$ for the 2 possible alignments between $o^1$ and $s^1$
- use $a^2_1, a^2_2$ for the 2 possible alignments between $o^2$ and $s^2$

The table below shows translations probabilities, with $tr(o|s)$ shown at row $o$, col $s$, and they are all initialised to $\frac{1}{3}$

| $tr(o|s)$ | $ca$ | $ve$ | $la$ |
|-----------|------|------|------|
| $gr$      | $\frac{1}{3}$ | $\frac{1}{3}$ | $\frac{1}{3}$ |
| $ho$      | $\frac{1}{3}$ | $\frac{1}{3}$ | $\frac{1}{3}$ |
| $the$     | $\frac{1}{3}$ | $\frac{1}{3}$ | $\frac{1}{3}$ |

To execute the brute force EM algorithm we need first for the pairs $o^1, s^1$ and $o^2, s^2$ to determine the conditional alignment probabilities so $p(a|o^1, s^1)$ and $p(a|o^2, s^2)$. The slides gave a derivation of the formula for $p(a|o, s)$, it came out to be

$$p(a|o, s) = \frac{\prod_j [p(o_j|s_{a(j)})]}{\sum_{a'} \prod_j [p(o_j|s_{a'(j)})]}$$  \hspace{1cm} (1)

and in the derivation $\frac{1}{(3+3)_2}$ terms cancelled. Even though for this assignment we are restricting the space of assignments the same formula applies – its applicability just depends on the value $p(a|t_o, s)$ being a uniform distribution\(^1\).

As a name for the numerator term in (1) we will use $num(a)$

So to determine the $p(a|o, s)$ values for each pair we need to

\(^1\)Numerically for each $a$ the term is $\frac{1}{3^3}$
1. for each possible \( a \) determine \( \text{num}(a) \) (ie. \( \prod_{j}[p(o_j|s_{a(j)})] \))

2. sum these to give the denominator \( \sum_{a} \text{num}(a) \) and then take ratios

Armed with these conditional probabilities can then compute expected counts of \( o, s \) combinations across the corpus, and from these recalculate \( tr(o|s) \) probabilities.

**ITERATION 1**

considering the first pair, for each \( a^1_n \) calculate \( \text{num}(a^1_n) \):

\[
\text{num}(a^1_1) = \frac{1}{9},
\text{num}(a^1_2) = \frac{1}{9},
\text{ditto}
\]

sim. for each \( a^2_n \) calculate \( \text{num}(a^2_n) \). At this stage, these all work out as \( \frac{1}{9} \).

from these to calculate the conditional probabilities \( P(a^d_n|o, s) \), need to sum the \( \text{num}(a^n) \) by summing across the table and use it as denominator ie.

\[
P(a^d_n|o, s) = \frac{\text{num}(a^d_n)}{\sum_{n'} \text{num}(a_{n'}^d)}
\]

\[
P(a^1_1|o^1, s^1) = \frac{1/9}{1/9} = \frac{1}{9},
\text{ditto}
\]

\[
P(a^1_2|o^1, s^1) = \frac{1/9}{1/9} = \frac{1}{9},
\text{ditto}
\]

\[
P(a^2_1|o^2, s^2) = \frac{1/9}{1/9} = \frac{1}{9},
\text{ditto}
\]

\[
P(a^2_2|o^2, s^2) = \frac{1/9}{1/9} = \frac{1}{9}.
\]

Notice these numbers make intuitive sense: with all \( tr(o|s) \) set equal, all alignments should be equally probable, giving a value of \( \frac{1}{9} \) for each.

Now for each possible vocabulary combination \( o, s \) combination we have to make a count by going through all the alignments and incrementing the count by how many times \( o \) is paired with \( s \) in the alignment and multiplying that by the above conditional alignment probabilities

For these short sentences the \( o, s \) count for any alignment is at most 1, and it will be handy for the calculations to note for each \((o, s)\) the alignments where it occurs once:\n
<p>| | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>ca</td>
<td>gr</td>
<td>ho</td>
<td>the</td>
</tr>
<tr>
<td></td>
<td>1:1</td>
<td>1:2</td>
<td>1:-</td>
</tr>
<tr>
<td>2:-</td>
<td>2:1</td>
<td>2:2</td>
<td></td>
</tr>
<tr>
<td>ve</td>
<td>1:2</td>
<td>1:1</td>
<td>1:-</td>
</tr>
<tr>
<td>2:-</td>
<td>2:-</td>
<td>2:-</td>
<td></td>
</tr>
<tr>
<td>la</td>
<td>1:-</td>
<td>1:-</td>
<td>1:-</td>
</tr>
<tr>
<td>2:-</td>
<td>2:2</td>
<td>2:1</td>
<td></td>
</tr>
</tbody>
</table>

based on this we get the following expected counts

\(^2\)to read this table the (ve,gr) entry has 1/2 to indicate in first pair \((o^1, s^1)\), the (ve,gr) pairing occurs in alignment \( a^1_2 \), and the pairing never occurs in the alignments for the second pair
\[
\begin{array}{cccc}
cat & gr & ho & the \\
ca  & 1 \times \frac{1}{2} & 2 \times \frac{1}{2} & 1 \times \frac{1}{2} \\
ve  & 1 \times \frac{1}{2} & 1 \times \frac{1}{2} & 0 \\
la  & 0 & 1 \times \frac{1}{2} & 1 \times \frac{1}{2} \\
\end{array}
\]

and for these counts get new \( tr(o|s) \) by normalising by column sums

\[
\begin{array}{cccc}
tr(o|s) & gr & ho & the \\
ca  & \frac{1}{2} & \frac{1}{2} & \frac{1}{2} \\
ve  & \frac{1}{2} & \frac{1}{2} & 0 \\
la  & 0 & \frac{1}{2} & \frac{1}{2} \\
\end{array}
\]

You should complete two further iterations in the same fashion.