CS/ENGRI 172, Fall 2002
11/22/02: Lecture Thirty-Six Handout

Topics: Statistical machine translation: the IBM Candide system. Our presentation roughly follows Kevin Knight's (1999) tutorial.

Conventions and notation

For simplicity, we only consider word-for-word translations, with no insertions or deletions of words allowed. The algorithm itself is intuitively quite simple, but to describe it formally we need to build up some notation.

We use $\text{tr}(s \rightarrow t)$ to denote the *translation weight* (probability) that source-language word s should be translated as target-language word t.

The key idea is to consider an auxiliary source of information: *alignments* in sentence pairs made up of mutual translations. Let $p^{(1)}, p^{(2)}, \ldots, p^{(N)}$ be the source/target-language sentence pairs in the training corpus. Let the ith sentence pair be $p^{(i)} = (s_1^{(i)} s_2^{(i)} \cdots s_{\ell_i}^{(i)}; t_1^{(i)} t_2^{(i)} \cdots t_{\ell_i}^{(i)})$, where $s_1^{(i)} \cdots s_{\ell_i}^{(i)}$ is an ℓ_i-word source-language sentence and $t_1^{(i)} \cdots t_{\ell_i}^{(i)}$ is its ℓ_i-word translation.[1] Note that the $s_j^{(i)}$s don't have to be distinct, and similarly for the target-sentence words. Then, an alignment lists, for each of the ℓ_i words in the source-language sentence, which word of the target sentence it is aligned to, and thus takes the form

$$(1 \leftrightarrow j_1; 2 \leftrightarrow j_2; \cdots; \ell_i \leftrightarrow j_{\ell_i})$$

where the j_k's are all distinct numbers between 1 and ℓ_i inclusive. For example, the two pictures shown here:

correspond to the two alignments $(1 \leftrightarrow 1; 2 \leftrightarrow 3; 3 \leftrightarrow 2; 4 \leftrightarrow 4; 5 \leftrightarrow 5)$ and $(1 \leftrightarrow 4; 2 \leftrightarrow 2; 3 \leftrightarrow 5; 4 \leftrightarrow 1; 5 \leftrightarrow 3)$ respectively.

Each p_i has a set of m_i associated alignments[2] $A_1^{(i)}, A_2^{(i)}, \ldots, A_{m_i}^{(i)}$. Using the variable A to stand for an alignment drawn from an arbitrary sentence pair, we say that every alignment A has an *alignment weight* (probability) $\text{awt}(A)$.

We use the notation $\text{Contains}(s \leftrightarrow t)$ to denote the set of alignments in which source-language word s is aligned with target-language word t. Note that $\text{Contains}(s \leftrightarrow t)$ can include alignments from different sentence pairs. Let $\text{freq}(s \leftrightarrow t, A)$ be the number of times we have the word s aligned to t in alignment A.

(OVER)

[1] By our assumption above, the two sentences in any pair are the same length; we have the subscript i because sentences in *different* pairs can have different lengths.

[2] In fact, $m_i = (\ell_i)(\ell_i - 1)(\ell_i - 2) \cdots (2)(1)$.

An iterative learning algorithm for MT

1. Initialization: For every sentence pair p_i, set $\text{awt}(A_1^{(i)}) = \text{awt}(A_2^{(i)}) = \cdots = \text{awt}(A_{m_i}^{(i)}) = 1/(m_i)$.

2. Repeat the following steps in order until no "significant" change:
3. Update translation weights: For every source/target word pair (s, t), change $\text{tr}(s \rightarrow t)$ to $\sum_{A \text{ in } \text{Contains}(s \leftrightarrow t)} \text{freq}(s \leftrightarrow t, A)\text{awt}(A)$.
4. Pseudo-normalize translation weights: Change each weight $\text{tr}(s \rightarrow t)$ to $\text{tr}(s \rightarrow t)/\sum_{t'} \text{tr}(s \rightarrow t')$.
5. Update alignment weights: For every $A_k^{(i)} = (1 \leftrightarrow j_1; 2 \leftrightarrow j_2; \cdots; \ell_i \leftrightarrow j_{\ell_i})$, change $\text{awt}(A_k^{(i)})$ to $\text{tr}(s_1 \rightarrow t_{j_1})\text{tr}(s_2 \rightarrow t_{j_2}) \cdots \text{tr}(s_{\ell_i} \rightarrow t_{j_{\ell_i}})$.
6. Pseudo-normalize alignment weights: For every alignment $A_k^{(i)}$, change $\text{awt}(A_k^{(i)})$ to $\text{awt}(A_k^{(i)})/\sum_{q=1}^{m_i} \text{awt}(A_q^{(i)})$.

Connections can be drawn to the hubs-and-authorities algorithm we considered earlier in the course.

Example Suppose we have two sentence pairs, $p_1 = (\textit{chat bleu}; \textit{blue cat})$ and $p_2 = (\textit{chat}; \textit{cat})$. This yields three alignments:

$$\begin{aligned} A_1^{(1)} &= (1 \leftrightarrow 1; 2 \leftrightarrow 2) \quad (\text{so "}\textit{chat}\text{" aligned to "}\textit{blue}\text{"}) \\ A_2^{(1)} &= (1 \leftrightarrow 2; 2 \leftrightarrow 1) \quad (\text{so "}\textit{chat}\text{" aligned to "}\textit{cat}\text{"}) \\ A_1^{(2)} &= (1 \leftrightarrow 1) \quad (\text{only one possible choice}) \end{aligned}$$

		$\text{awt}(A_1^{(1)})$	$\text{awt}(A_2^{(1)})$	$\text{awt}(A_1^{(2)})$	$\text{tr}(\textit{chat} \rightarrow \textit{blue})$	$\text{tr}(\textit{chat} \rightarrow \textit{cat})$	$\text{tr}(\textit{bleu} \rightarrow \textit{blue})$	$\text{tr}(\textit{bleu} \rightarrow \textit{cat})$
a.	Init	1/2	1/2	1	–	–	–	–
b.	Up-tr	1/2	1/2	1	1/2	3/2	1/2	1/2
c.	PNorm-tr	1/2	1/2	1	1/4	3/4	1/2	1/2
d.	Update-a	1/8	3/8	3/4	1/4	3/4	1/2	1/2
e.	PNorm-a	1/4	3/4	1	1/4	3/4	1/2	1/2
f.	Update-tr	1/4	3/4	1	1/4	7/4	3/4	1/4
g.	Pnorm-tr	1/4	3/4	1	1/8	7/8	3/4	1/4