WORD ALIGNMENT MODELS

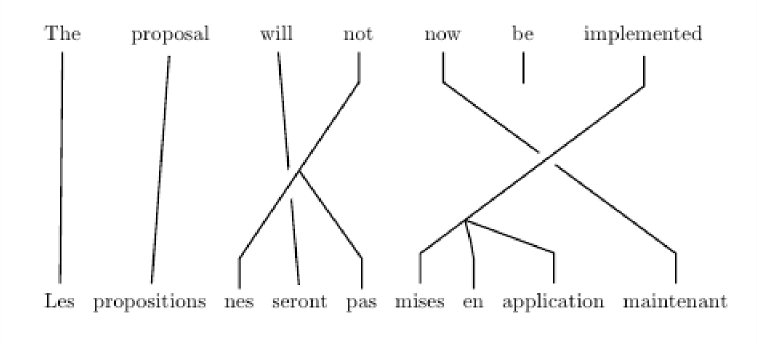
David Talbot

22nd April 2017

Computer Science Club, St. Petersburg, Russia

WORD ALIGNMENT MODELS

an alignment



‘The Mathematics of Machine Translation: Parameter Estimation‘, Brown et al. (1993).

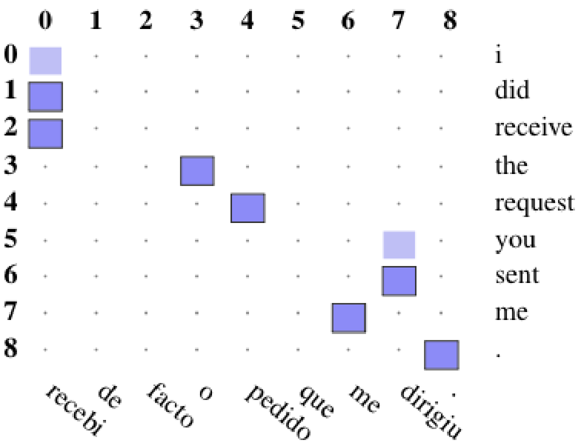
ibm papers (1990-1993)

- Formulated a generative model of parallel sentence pairs

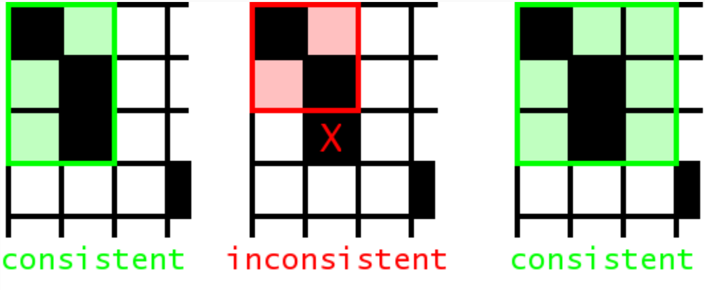
$$\Pr(F = \mathbf{f} | E = \mathbf{e}) = \sum_{\mathbf{a} \in \mathcal{A}} \Pr(A = \mathbf{a}, F = \mathbf{f} | E = \mathbf{e})$$

  where F is a French sentence, E is an English sentence and $\mathcal{A}$ is the set of all possible alignments for the sentence pair.
- Proposed using the EM algorithm to learn the parameters and infer word alignment matrix.

word alignment matrix



Natural way to visualize an alignment.



Word alignments constrain the set of possible phrase pairs.

aligning words in a parallel corpus

We're given corpus of translated sentence pairs $D = \{(\mathsf{e}, \mathsf{f})_1, (\mathsf{e}, \mathsf{f})_2, (\mathsf{e}, \mathsf{f})_3, ...\}$.

We assume these sentence pairs are distributed *i.i.d.* given the parameters θ,

$$\begin{aligned}
\Pr(D|\theta) &= \prod_{k \in D} \Pr(\mathsf{f}_k | \mathsf{e}_k, \theta) \\
&= \prod_{k \in D} \sum_{\mathsf{a}_k \in \mathcal{A}} \Pr(\mathsf{a}_k, \mathsf{f}_k | \mathsf{e}_k, \theta) \\
&= \prod_{k \in D} \sum_{\mathsf{a}_k \in \mathcal{A}} \underbrace{\Pr(\mathsf{a}_k | \mathsf{e}_k, \theta)}_{\text{Prior}} \underbrace{\Pr(\mathsf{f}_k | \mathsf{e}_k, \mathsf{a}_k, \theta)}_{\text{Translation model}}
\end{aligned}$$

Bias-variance trade-off

Simple models (few parameters) generalize better to new data, but may not capture the structure of the data (e.g. unigram n-gram model).

Complex models (many parameters) capture the structure of the training data, but generalize less well to new data (e.g. unsmoothed 5-gram model).

How do hidden variables complicate the choice of model structure?

choosing a model: hidden data

How does the structure of A (the alignments) affect the computation?

How big is A for a single sentence pair $|\mathbf{e}| = I$ and $|\mathbf{f}| = J$?

choosing a model: hidden data

How does the structure of A (the alignments) affect the computation?

How big is A for a single sentence pair $|\mathbf{e}| = I$ and $|\mathbf{f}| = J$?

$$\Pr(f_1, ..., f_J | e_1, ..., e_I, \theta) = \sum_{a_1=1}^{I} ... \sum_{a_J=1}^{I} \Pr(a_1, ..., a_J, f_1, ..., f_J | e_1, ..., e_I, \theta)$$

choosing a model: hidden data

How does the structure of A (the alignments) affect the computation?

How big is A for a single sentence pair $|\mathbf{e}| = I$ and $|\mathbf{f}| = J$?

$$\Pr(f_1, ..., f_J | e_1, ..., e_I, \theta) = \sum_{a_1=1}^{I} ... \sum_{a_J=1}^{I} \Pr(a_1, ..., a_J, f_1, ..., f_J | e_1, ..., e_I, \theta)$$

Exact E-step is only tractable for a very limited set of models.

simplifying assumptions

Assumption 1

Each French word f_j is generated independently given the English word to which it is aligned e_{a_j}

$$\Pr(\mathbf{f}|\mathbf{e}) \approx \prod_{j=1}^{I} \sum_{\mathbf{a} \in \mathcal{A}} \Pr(\mathbf{a}|\mathbf{e}, \theta) \Pr(f_j|e_{a_j}, \theta)$$

What's an obvious problem with this assumption?

simplifying assumptions

Assumption 2

We'll parameterize the translation model $\Pr(f_j | e_{a_j}, \theta)$ with a table of conditional probabilities $t(f|e)$.

E.g. for Russian to English translation the table $t(\mathrm{f}|dog)$ could be defined as

$$t(\text{собака}|\text{dog}) = 0.5$$

$$t(\text{собаку}|\text{dog}) = 0.3$$

$$t(\text{кошка}|\text{dog}) = 0.2.$$

What's an obvious problem with this?

simplifying assumptions

Assumption 3

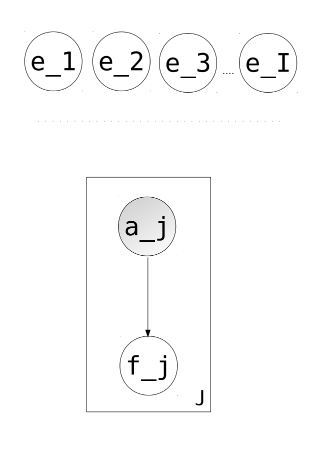
We'll simplify the 'prior' $\mathsf{Pr}(a|\mathbf{e},\theta)$ significantly.

At first we'll assume a uniform prior, i.e. that all alignments are *a priori* equally likely (i.e. they don't depend on the English words or any other alignments).

$$\forall \mathbf{a} \in A, \mathsf{Pr}(\mathbf{a}|\mathbf{e},\theta) = \epsilon.$$

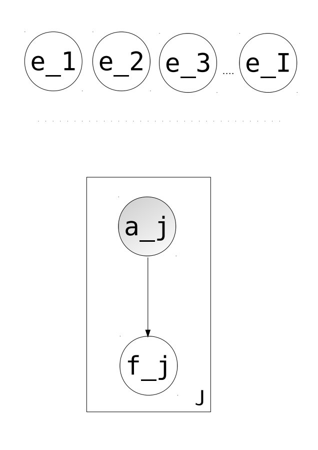
Why is this not a great assumption?

ibm model 1



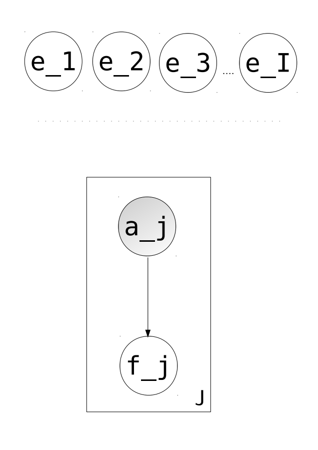
$$\Pr(\mathsf{f}, \mathsf{a} | \mathsf{e}, \theta) \approx \prod_{j=1}^{J} \Pr(f_j, a_j | \mathsf{e}, \theta)$$

ibm model 1



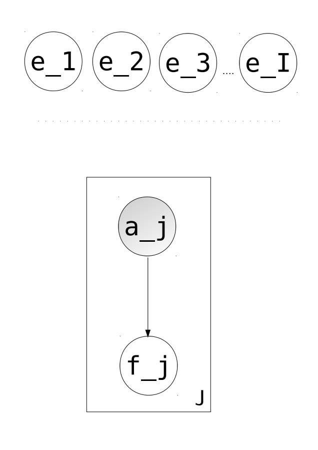
$$
\begin{aligned}
\Pr(\mathsf{f}, \mathsf{a} | \mathsf{e}, \theta) &\approx \prod_{j=1}^{J} \Pr(f_j, a_j | \mathsf{e}, \theta) \\
&= \prod_{j=1}^{J} \Pr(a_j | \mathsf{e}) \Pr(f_j | \mathsf{e}, a_j, \theta)
\end{aligned}
$$

ibm model 1



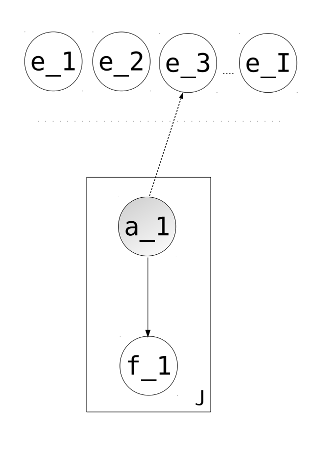
$$
\begin{aligned}
\Pr(\mathsf{f}, \mathsf{a} | \mathsf{e}, \theta) &\approx \prod_{j=1}^{J} \Pr(f_j, a_j | \mathsf{e}, \theta) \\
&= \prod_{j=1}^{J} \Pr(a_j | \mathsf{e}) \Pr(f_j | \mathsf{e}, a_j, \theta) \\
&\approx \prod_{j=1}^{J} \epsilon \Pr(f_j | e_{a_j}, \theta)
\end{aligned}
$$

ibm model 1



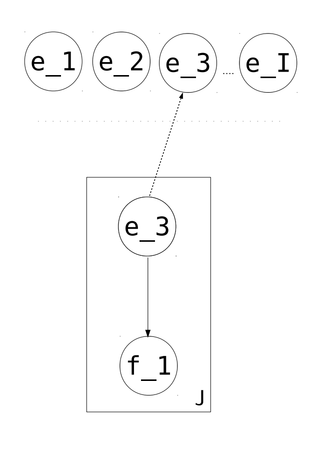
$$
\begin{aligned}
\Pr(\mathbf{f}, \mathbf{a} | \mathbf{e}, \theta) &\approx \prod_{j=1}^{J} \Pr(f_j, a_j | \mathbf{e}, \theta) \\
&= \prod_{j=1}^{J} \Pr(a_j | \mathbf{e}) \Pr(f_j | \mathbf{e}, a_j, \theta) \\
&\approx \prod_{j=1}^{J} \epsilon \Pr(f_j | e_{a_j}, \theta) \\
&\propto \prod_{j=1}^{J} t(f_j | e_{a_j})
\end{aligned}
$$

ibm model 1

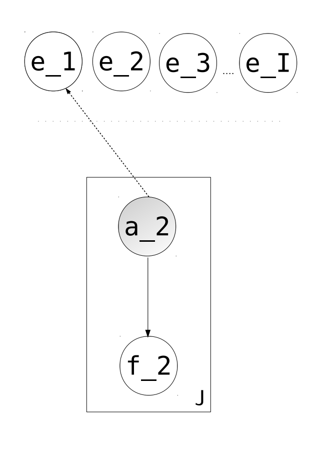


$$
\begin{aligned}
\Pr(\mathrm{f}, \mathrm{a} | \mathrm{e}, \theta) &\approx \prod_{j=1}^{J} \Pr(f_j, a_j | e_{a_j}, \theta) \\
&= \Pr(f_1, a_1 = 3 | e_3, \theta) \ldots
\end{aligned}
$$

ibm model 1

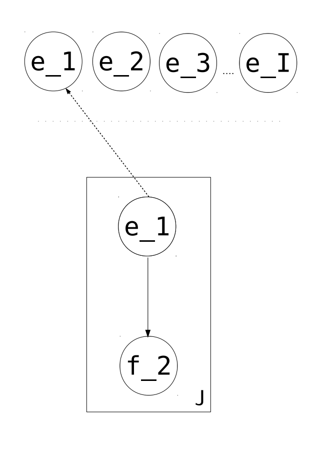


$$
\begin{aligned}
\Pr(\mathrm{f}, \mathrm{a} | \mathrm{e}, \theta) &\approx \prod_{j=1}^{J} \Pr(f_j, a_j | e_{a_j}, \theta) \\
&= \Pr(f_1, a_1 = 3 | e_3, \theta) \ldots \\
&\approx t(f_1, | e_3) \ldots
\end{aligned}
$$



$$
\begin{aligned}
\Pr(\mathrm{f}, \mathrm{a} | \mathrm{e}, \theta) &\approx \prod_{j=1}^{J} \Pr(f_j, a_j | e_{a_j}, \theta) \\
&= \Pr(f_1, a_1 = 3 | e_3, \theta) \ldots \\
&\approx t(f_1, | e_3) \ldots
\end{aligned}
$$

ibm model 1



$$
\begin{aligned}
\Pr(\mathrm{f}, \mathrm{a} | \mathrm{e}, \theta) &\approx \prod_{j=1}^{J} \Pr(f_j, a_j | e_{a_j}, \theta) \\
&= \Pr(f_1, a_1 = 3 | e_3, \theta) \dots \\
&\approx t(f_1, | e_3) \dots \\
&\approx t(f_1, | e_3) t(f_2 | e_1) \dots
\end{aligned}
$$

The expected log-likelihood for $\mathbf{f}$ given $\mathbf{e}$ under IBM Model 1 is

$$\mathbb{E}[log(\mathbf{f}|\mathbf{e},\theta)] = \sum_{j=1}^{J}\sum_{i=1}^{I} \Pr(a_j = i|\mathbf{f},\mathbf{e},\theta) \log \Pr(f_j, a_j = i|e_i, \theta)$$
$$\propto \sum_{j=1}^{J}\sum_{i=1}^{I} \Pr(a_j = i|\mathbf{f},\mathbf{e},\theta) \log t(f_j|e_i).$$

To apply EM we need to compute $\Pr(a_j = i|\mathbf{f},\mathbf{e},\theta)$ for each source and target pair and then maximize this term w.r.t. our parameters $\theta = t(f|e)$.

em for ibm model 1 cont.

The posterior alignment probabilities, $\Pr(a_j = i|\mathbf{f}, \mathbf{e}, \theta)$ can be computed as follows

$$\begin{aligned}
\Pr(a_j = i|\mathbf{f}, \mathbf{e}, \theta) &= \frac{\Pr(f_j, a_j = i|\mathbf{e}, \theta)}{\Pr(f_j|\mathbf{e}, \theta)} \\
&= \frac{\Pr(a_j = i|\mathbf{e}, \theta)\Pr(f_j|a_j = i, \mathbf{e}, \theta)}{\sum_{k=1}^{I} \Pr(a_j = k|\mathbf{e}, \theta)\Pr(f_j|a_j = k, \mathbf{e}, \theta)} \\
&= \frac{\epsilon t(f_j|e_i)}{\sum_{k=1}^{I} \epsilon t(f_j|e_k)} \\
&= \frac{t(f_j|e_i)}{\sum_{k=1}^{I} t(f_j|e_k)}.
\end{aligned}$$

measuring alignment quality

Given a golden set of manually created M consisting of probable P and sure S alignments. We can measure the error rate of an automatic alignment A:

$$Precision(A;P) = \frac{|P \cap A|}{|A|}$$

$$Recall(A;S) = \frac{|S \cap A|}{|S|}$$

$$AlignmentErrorRate(A;S,P) = 1 - \frac{|P \cap A| + |S \cap A|}{|S| + |A|}.$$

improving on model 1

Suggestions:

- Parameter tying
- Better use of positional information in prior (e.g. words align with words close by)
- Using prior information, e.g. character level model
- Using linguistic annotations (see other files)

Who can get the lowest *alignment error rate* by tomorrow?