HAL: Challenging Three Key Aspects of IBM-style Statistical Machine Translation

Christer Samuelsson
UiL-OTS
Utrecht University
a.c.j.samuelsson@uu.nl

Abstract

The IBM schemes use weighted cooccurrence counts to iteratively improve translation and alignment probability estimates. We argue that: 1) these cooccurrence counts should be combined differently to capture word correlation; 2) alignment probabilities adopt predictable distributions; and 3) consequently, no iteration is needed. This applies equally well to word-based and phrase-based approaches. The resulting scheme, dubbed HAL, outperforms the IBM scheme in experiments.

1 Introduction

Statistical machine translation views a source-language text as a corrupted version of an unknown target-language text. The translation task is thus one of reverse decoding: it consists in finding the most likely target-language text from which a given source-language text was generated.

$$\underset{\textit{Target}}{\operatorname{argmax}} \; P(\textit{Target} \mid \textit{Source}) \;=\; \underset{\textit{Target}}{\operatorname{argmax}} \; P(\textit{Source} \mid \textit{Target}) \cdot P(\textit{Target})$$

The fundamental translation units are sentences and words: sentences are viewed as finite strings of words. In statistical machine translation, the term phrases refers to substrings of words.

The translation model relies on the bilexical[1] probabilities $P(t \rightsquigarrow s)$ that a target-language word t generates a source-language word s, and the probabilities $P_{\mathrm{SL}}(J \mid I)$ that a target sentence of length I generates a source sentence of length J.

This model also uses intra-sentence alignment probabilities, relating the word order in the target and source sentences. $P(i \mapsto j \mid \bullet)$—often written $P(a_j = i \mid \bullet)$—indicates the probability that the target word in position i (irrespective of what this word might be) generates the source word in position j (irrespective of what that word might be), given the context $\bullet$. Here, j may be zero, indicating that the target word in position i has no realization in the source sentence. We let the the context $\bullet$ be i, I, J, resulting in IBM model 2 (Brown et al., 1993).

IBM models use weighted cooccurrence counts to improve bilexical and alignment probability estimates iteratively, as described in Section 2. In Section 3 we argue that the resulting weighted cooccurrence counts should be combined differently to capture word correlation. In Sections 4 and 5 we argue and show empirically that alignment probabilities adopt predictable distributions, and, in Section 6, that the iterative procedure can be eliminated. The experiments of Section 7 support our claims.

These results reach far beyond the word-based IBM model 2 used in the experiments. They reveal how **correlation-weighted translation probabilities** *capture word correlation, and how* **parametric distributions** *can replace latent alignment probabilities created by the IBM scheme. This allows* **eliminating iteration**. *We urge readers, for whom phrase-based translation is paramount, to mentally perform the substitutions of Section 9:*

$$t \;\rightarrow\; \tau \quad ; \quad s \;\rightarrow\; \sigma \quad ; \quad i \;\rightarrow\; \{i_l, i_h\}$$

[1] Our analysis works equally well for biphrasal probabilities $P(\tau \rightsquigarrow \sigma) = P(t_1 \ldots t_m \rightsquigarrow s_1 \ldots s_n)$ and alignment probabilities $P(\{i_l, i_h\} \mapsto \{j_l, j_h\} \mid \bullet)$, see Section 9. In fact, continuous distributions simplify calculating the latter.

2 The IBM Scheme

In which we recapitulate the IBM update cycle.

In this article, we let $P(\bullet)$ denote the actual probability of an event $\bullet$, let $p(\bullet)$ denote a parameter estimate, and let $\tilde{p}(\bullet)$ denote an empirical probability (normally a relative frequency, but here the latent distributions generated by the IBM scheme: we reserve $p(\bullet)$ for other types of estimates). It is a fundamental assumption of statistics that actual probabilities exist as limits of relative frequencies for sufficiently large numbers of observations. (Bayesian statisticians envision also other types of actual probabilities.)

In the IBM model 2 update cycle (Brown et al., 1993), pp. 273–274, one improves the estimates $\tilde{p}(t \rightsquigarrow s)$ and $\tilde{p}(i \mapsto j \mid i, I, J)$ of the bilexical and alignment probabilities, respectively, by iteration.

For each position pair $\langle i, j \rangle \in I_k \times J_k$ of each sentence pair $\langle t_1^k \ldots t_{I_k}^k ; s_1^k \ldots s_{J_k}^k \rangle$, one weights the cooccurrence counts with the product of the previous estimates of these probabilities.

$$\begin{aligned} p_k(i,j) &\equiv \tilde{p}(t_i^k \rightsquigarrow s_j^k) \cdot \tilde{p}(i \mapsto j \mid i, I_k, J_k) \\ p_k(i \mid j) &\equiv \frac{p_k(i,j)}{\sum_{i'} p_k(i',j)} \end{aligned}$$

The weighted cooccurrence counts are

$$\begin{aligned} A(i,j,I,J) &= \sum_k p_k(i \mid j) \cdot \delta_{I,I_k} \cdot \delta_{J,J_k} \\ A(i,I,J) &= \sum_j A(i,j,I,J) \\ B(t,s) &= \sum_k \sum_{i,j} p_k(i \mid j) \cdot \delta_{t,t_i^k} \cdot \delta_{s,s_j^k} \\ B(t) &= \sum_s B(t,s) \\ \delta_{X,Y} &= \begin{cases} 1 & \text{if } X = Y \\ 0 & \text{otherwise} \end{cases} \end{aligned}$$

where $\delta_{X,Y}$ is the Kronecker delta, which does the actual counting.

The improved probability estimates are then

$$\begin{aligned} \tilde{p}(i \mapsto j \mid i, I, J) &\leftarrow \frac{A(i,j,I,J)}{A(i,I,J)} \\ \tilde{p}(t \rightsquigarrow s) &\leftarrow \frac{B(t,s)}{B(t)} \end{aligned}$$

We take issue with these estimates in Section 3.

Since the unknown bilexical and the alignment probabilities occur both in the left-hand and right-hand sides, the reasoning goes, one must resort to iteration. We challenge this premise in Sections 4 and 5, and its conclusion in Section 6. One starts with some initial sets of probabilities, e.g., uniform distributions, and iterates to a self-consistent parameter setting. This is an instance of the EM algorithm, which is very well described in (Bishop, 2006), pp. 424–455.

The IBM scheme can be viewed as an optimizer. Given a set of translation and alignment probabilities, it produces a better set of such probabilities. Better here means increasing the likelihood of the training data under the model in question. It is crucial to understand that there is no guarantee that this will result in a better model as measured by other criteria, such as improved translation quality.

IBM model 2 is typically not deployed in itself. It often provides a set of bilexical probabilities that are used to seed more elaborate IBM models, or to extract biphrasal probabilities or lexical features for generative and non-generative translation models.

3 Word Correlation

In which we argue that bilexical probability estimates be proportional to $(D'(t,s) - \lambda) \cdot \frac{C_w(t,s)}{C_w(t)}$.

Consider a general weight $w_k(i,j)$ when counting word cooccurrences.

$$C_w(t,s) = \sum_k \sum_{i,j} w_k(i,j) \cdot \delta_{t,t_i^k} \cdot \delta_{s,s_j^k}$$

Using the weights $p_k(i \mid j)$ of Section 2 yields the weighted counts $C_w(t,s) = B(t,s)$ of the IBM scheme. If all weights equal one, then $C_w(t,s) = C(t,s)$ is the number of times t and s cooccurred in the data.

A very intuitive measure of word correlation is the Dice coefficient (Dice, 1945),[2]

$$D(t,s) = \frac{C(t,s)}{C(t) \cdot C(s)}$$

[2] We have omitted a factor two from the expression.

which is closely related to the mutual information statistics between two random variables X and Y.

$$I(X,Y) \equiv \mathrm{E}\left[\ln\left(\frac{P(X,Y)}{P(X)\cdot P(Y)}\right)\right]$$

Many statistical machine translation approaches, e.g., (Smadja et al., 1996), are based on the Dice coefficient. Although not the original intention, we extend it to weighted coocurrence counts $C_w(\bullet)$.

3.1 Dimensional Analysis

We apply dimensional analysis (Hornung, 2006) to probabilities $P_w(\bullet)$ and frequency counts $C_w(\bullet)$.

$$\begin{aligned}\dim[C_w(\bullet)] &\equiv \dim[w]\cdot\dim[\bullet]\\ \dim[P_w(\bullet)] &\equiv \dim\left[\frac{C_w(\bullet)}{C_w()}\right] = \dim[\bullet]\\ \dim[P_w(X\mid Y)] &\equiv \dim\left[\frac{P_w(X,Y)}{P_w(Y)}\right] = \dim[X]\end{aligned}$$

$C_w()$ is the total (weighted) count and it has dimension W.

For the bilexical counts we obtain

$$\begin{aligned}\dim[C_w(t)] &= \dim[w]\cdot\dim[t] \equiv W\cdot T\\ \dim[P_w(s\mid t)] &= \dim[s] \equiv S\end{aligned}$$

We thus want an estimate of $P(t\rightsquigarrow s)$ with dimension S. Under IBM models, the bilexical probability estimates have dimension S, as they should.

$$\dim[\tilde{p}(t\rightsquigarrow s)] = \dim\left[\frac{B(t,s)}{B(t)}\right] = S$$

The Dice coefficient $D_w(t,s)$ has dimension $\frac{1}{W}$.

$$\begin{aligned}&\dim[D_w(t,s)] =\\ &= \frac{\dim[C_w(t,s)]}{\dim[C_w(t)]\cdot\dim[C_w(s)]} = \frac{1}{W}\end{aligned}$$

We render the Dice coefficient dimensionless by multiplying it with the total count $C_w()$, which has dimension W.

$$D'(t,s) = \frac{C_w(t,s)\cdot C_w()}{C_w(t)\cdot C_w(s)} = \frac{P_w(t,s)}{P_w(t)\cdot P_w(s)}$$

This means that the mutual information is

$$I(X,Y) \equiv \mathrm{E}\left[\ln D'(X,Y)\right]$$

If $D'(t,s)<1$, then $\ln D'(t,s)<0$, and t and s are negatively correlated, and $p(t\rightsquigarrow s)$ should be zero. Note that for $D'(t,s)\approx 1$, we have

$$\ln D'(t,s) = \ln(1+D'(t,s)-1) \approx D'(t,s)-1$$

Assume that our estimate of $P(t\rightsquigarrow s)$ is a function of $C_w(t,s)$, $C_w(t)$, $C_w(s)$, and $C_w()$ and let S,T,W be our standard dimensions. We can then use the (modified) Dice coefficient $D'(t,s)$ as our single dimensionless variable: all possible estimates of dimension S are then of the form

$$p(t\rightsquigarrow s) \propto g(D'(t,s))\cdot\frac{C_w(t,s)}{C_w(t)}$$

for some function $g(D'(t,s))$.

The most obvious choice is $g(D'(t,s))=1$. This is the one used in Section 2.

$$\tilde{p}(t\rightsquigarrow s) = \frac{B(t,s)}{B(t)} = 1\cdot\frac{C_w(t,s)}{C_w(t)}$$

This formula has the correct dimension, but it does not measure word correlation, being independent of $C_w(s)$; nor is it zero for $D'(t,s)=1$.

We instead use $g(D'(t,s)) = D'(t,s)-\lambda$, arriving at

$$\begin{aligned}p(t\rightsquigarrow s) &\propto (D'(t,s)-\lambda)\cdot\frac{C_w(t,s)}{C_w(t)} =\\ &= \left(\frac{C_w(t,s)\cdot C_w()}{C_w(t)\cdot C_w(s)}-\lambda\right)\cdot\frac{C_w(t,s)}{C_w(t)}\end{aligned}$$

where λ is a real parameter. This is the simplest function of $C_w(t,s)$, $C_w(t)$, $C_w(s)$, and $C_w()$ with the correct dimension, that measures word correlation and is, with $\lambda=1$, zero for $D'(t,s)=1$. We set $p(t\rightsquigarrow s)=0$ for $D'(t,s)<1$.

In the following, we will use $\lambda=0$, rather than $\lambda=1$, to get a more inclusive bilexon.

3.2 Null Words

Unlike the IBM scheme, we do not collect weighted counts for a hypothetical null word ϵ. Had we done so, with a fixed null word alignment probability p_0, their Dice coefficients would be about one.

$$D'(t,\epsilon) \approx D'(\epsilon,s) \approx 1$$

since, theoretically,[3]

$$C_w(t,\epsilon) = C(t)\cdot p_0 \quad ; \quad C_w(t) = C(t)$$
$$C_w(\epsilon) = C()\cdot p_0 \quad ; \quad C_w() = C()$$

Instead, we set the deletion and insertion probabilities $p(t \rightsquigarrow \epsilon)$ and $p(\epsilon \rightsquigarrow s)$, respectively, to

$$p(t \rightsquigarrow \epsilon) \propto p_0 \quad ; \quad p(\epsilon \rightsquigarrow s) \propto C(s)$$

before renormalizing and symmetrizing.

4 Parametric Distributions: Theory

In which we argue that latent alignment probabilities can be replaced with reflected t distributions.

It is often claimed that systematic language differences, such as pre- vs post-modifiers, auxiliary verbs, verb movement, etc., will influence alignment probabilities. We maintain that for as simple ones as $P(i \mapsto j|i,I,J)$, averaging over a large number of sentences, with sequences of a variety of such patterns in various positions, will crisscross these patterns until most traces of them are lost.

We will use reflected Student's t distributions. However, both the reflection scheme of Section 4.2 and the parameter estimates of Section 4.3 apply to any continuous distribution $f(x;\mu,\sigma)$ on $(-\infty,\infty)$.

4.1 Student's t Distribution

A random variable obeying a Student's t distribution with r degrees of freedom has a probability density function—a pdf—$f(x)$ as follows.

$$f(x;\mu,\sigma,r) = \\ = \frac{\Gamma\left(\frac{r+1}{2}\right)}{\sqrt{\pi r\sigma^2}\Gamma\left(\frac{r}{2}\right)}\left(1+\frac{1}{r}\left(\frac{x-\mu}{\sigma}\right)^2\right)^{-\frac{r+1}{2}}$$

It has three parameters: μ, σ, and r. The special case of $r=1$ is known as the Cauchy distribution. The limiting distribution when $r\to\infty$ is the Gaussian:

$$f(x;\mu,\sigma) = \frac{1}{\sqrt{2\pi\sigma^2}}\exp\left(-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2\right)$$

[3] i.e., if, for all k (we then include $i=0$ and $j=0$)

$$\sum_i w_k(i,j) = \sum_j w_k(i,j) = 1$$

which is only approximately true with pruning and/or with our choice of $w(i,j)$, see the following section. Symmetrizing $w(i,j)$ is theoretical work in progress.

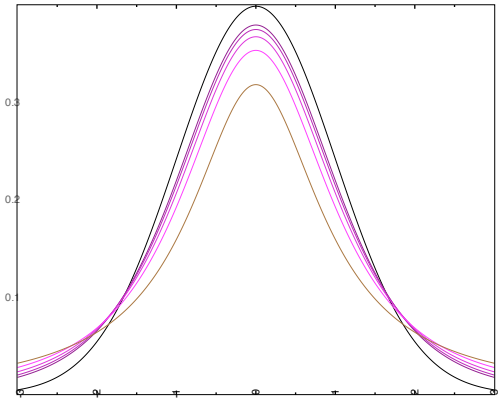

Figure 1: The pdf of Student's t distribution with $r = 1,\ldots,5$

For $r=1$ (the Cauchy distribution), neither the first nor second moment (expectations of x and x^2) exists, μ indicates the position of the single mode and σ is a scaling parameter. For $r=2$, the first moment exists and the expectation of x equals μ, but σ is still a scaling parameter. For $r>2$, the variance is $\frac{r}{r-2}\sigma^2$.

Figure 1 shows the pdfs of t distributions (in shades of purple) with $r = 1\ldots5$,[4] and a Gaussian distribution, all with the same parameters $\mu=0$ and $\sigma=1$. The t distributions have slimmer shoulders and fatter tails than the Gaussian, the economic ramifications of which have recently been manifesting globally (Taleb, 2007).

4.2 Reflection

We need to restrict the continuous distribution on $(-\infty,\infty)$ to $\{1,..J\}$. We first fold it into an interval $[L,H]$ by summing the contributions from distributions reflected in L and H, and their mirror images, ad infimum:

$$\hat{f}(x) = \sum_{k=-\infty}^{\infty} f(x+2kJ) + f(2L-x+2kJ)$$

Here $J = H-L$. Note that $2L+2J = 2H$.

Figure 2 shows a reflected Cauchy distribution, along with its mirror distributions (in brown). This physics-inspired scheme assigns higher probabilities towards the interval ends, than does simply renormalizing by the interval probability mass.

[4] The Cauchy distribution ($r=1$) is in brown.

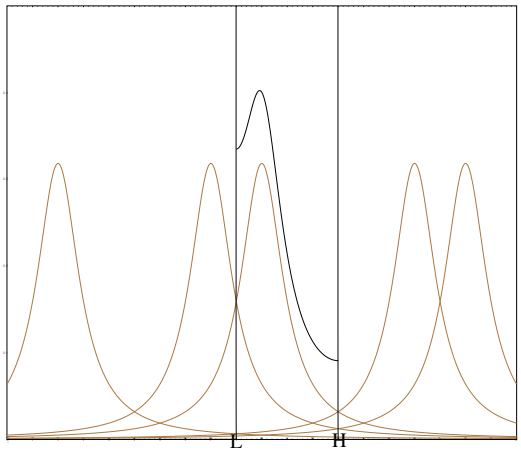

Figure 2: A reflected Cauchy distribution

We use this to apply parametric distributions to a source-language sentence of finite length J by reflecting (ad infimum) in $L = \frac{1}{2}$ and $H = J + \frac{1}{2}$. The sentence starts at $L = \frac{1}{2}$ and ends at $H = J + \frac{1}{2}$, since $J = H - L$ and $\frac{L+H}{2} = \frac{J+1}{2}$. Another way to think about this is that the interval $[1, J]$ has length $J - 1$, whereas the sentence length is J, so we need to pad it with $\frac{1}{2}$ in both ends, arriving at $[\frac{1}{2}, J + \frac{1}{2}]$.

We use the value of the pdf in integer points as probabilities. Theoretically, the numeric normalization factor differs for a sum and an integral, and a pdf may exceed one, but we may safely ignore this. In phrase-based approaches, it is practical to be able to use rational points or to integrate over subintervals.

4.3 Parameters

We need to estimate the parameters μ, σ, and r. In Section 5, we test $r = 1, 2, \ldots, 5$ and estimate the parameters μ_{iIJ} and σ_{iIJ} for each target sentence position i, and target and source sentence length I and J, from the latent alignment probabilities generated by the IBM scheme of Section 2.

Of course, first running the IBM scheme defeats the purpose of eliminating it, so we instead devise language-independent (L-I) parameters, *which turn out to work just as well as both the fitted parameters and the extracted latent alignment probabilities.*

To simplify the expressions, we introduce

$$\hat{\imath} = i - \frac{1}{2} \quad ; \quad \hat{\jmath} = j - \frac{1}{2}$$

$$p(i \mapsto j \mid i, I, J) = \sum_k f(\pm\hat{\jmath} + 2kJ; \mu_{\hat{\imath}IJ}, \sigma_{\hat{\imath}IJ}, r)$$

We then set the language-independent parameters to

$$\mu_{\hat{\imath}IJ} = \frac{\hat{\imath}}{I} J$$

$$\sigma^2_{\hat{\imath}IJ} = \frac{\hat{\imath}}{I}\left(1 - \frac{\hat{\imath}}{I}\right) \cdot J = \frac{\hat{\imath}(I - \hat{\imath})}{I^2} J$$

The modes of the beginning, middle, and end are—quite reasonably— the beginning, middle, and end.

$$\mu_{0IJ} = 0 \quad ; \quad \mu_{\frac{I}{2}IJ} = \frac{J}{2} \quad ; \quad \mu_{III} = J$$

These estimate are based on a Binomial distribution $\mathrm{Bin}(J, \frac{\hat{\imath}}{I})$ with J trials and probability $\frac{\hat{\imath}}{I}$. The Binomial distribution itself is however too close to a Gaussian. One could argue that the variance instead be proportional to J^2. We use as an analogy the difference between a deliberate and a random walk. The former will land us a distance from the starting point proportional to the walking time; the latter proportional to its square root.

5 Parametric Distributions: Experiments

In which we show empirically that the language-independent parameters work just as well as both the fitted parameters and the extracted latent alignment probabilities themselves.

We automatically fitted t distributions with 1, 2, 3, 4, and 5 degrees of freedom r to the latent alignment probabilities generated by the IBM scheme of Section 2. The first distribution, $r = 1$, is the Cauchy distribution; the last one, $r = 5$, is fairly close to the Gaussian around the mode, see Figure 1.

For each target sentence position i, and target and source sentence length I and J, the parameters μ_{iIJ} and σ_{iIJ} of the chosen distribution reflected only once in $\frac{1}{2}$ and $J + \frac{1}{2}$ were fitted automatically to the latent distribution $\tilde{p}(i \mapsto j \mid i, I, J)$. Null probabilities were omitted and the rest were renormalized.

$$p(i \mapsto j \mid i, I, J) = f(j; \mu_{iIJ}, \sigma_{iIJ}, r) + \\ + f(1\text{–}j; \mu_{iIJ}, \sigma_{iIJ}, r) + f(2J\text{+}1\text{–}j; \mu_{iIJ}, \sigma_{iIJ}, r)$$

5.1 Alignment Experiment 1

We calculated, for each target sentence position i, and target and source sentence length I and J, the

Kullback-Leibler (KL) divergence between the latent and fitted alignment distributions:

$$D_{KL}(\tilde{p}_{iIJ} \,||\, p_{iIJ}) \;=$$
$$= \sum_j \tilde{p}(i \mapsto j \mid i, I, J) \log \frac{\tilde{p}(i \mapsto j \mid i, I, J)}{p(i \mapsto j \mid i, I, J)}$$

The following table shows the average KL divergence to the latent distribution, weighted by the frequency count of $\langle I, J \rangle$, for $r = 1, \ldots, 5$. If this count was less than 100, its contribution was omitted to avoid data sparseness problems. We measure the KL divergence for both the fitted (Fit) and language-independent (L-I) parameters. In the last column (Uni), the average KL divergence to the latent distribution from a uniform distribution $p = \frac{1}{J}$ is given as a reference point.

Kullback-Leibler Divergence							
Lang	Para	Degrees of freedom (r)					Uni
		1	2	3	4	5	
Fin	Fit	0.088	0.064	0.077	0.092	0.105	0.77
	L-I	0.116	0.105	0.129	0.154	0.175	—
Fra	Fit	0.107	0.085	0.101	0.120	0.138	1.12
	L-I	0.140	0.123	0.146	0.170	0.193	—
Ger	Fit	0.106	0.095	0.117	0.138	0.156	0.86
	L-I	0.164	0.166	0.197	0.225	0.249	—
Swe	Fit	0.094	0.081	0.104	0.128	0.149	1.15
	L-I	0.150	0.136	0.161	0.187	0.211	—

Two degrees of freedom seems to be optimal. The KL divergences are less than 0.1, and less than a tenth (except, just barely, for German) of that of a uniform distribution. This is a very good fit indeed.

The KL divergences are low across the board, especially compared to that of a uniform distribution. This shows that both the fitted and language-independent distributions model the latent alignment probabilities found by the IBM scheme very well.

This in turn shows that the latent alignment probabilities contain less information than often assumed, confirming our intuition at the opening of Section 4 that averaging over a large number of sentences would crisscross language-pair-specific word-order patterns until most traces of them were lost.

5.2 Alignment Experiment 2

We visually inspected the language-independent, fitted, and latent alignment distributions for $I = J = 17$: $i = 2$, $i = 10$, and $i = 16$. The solid curves

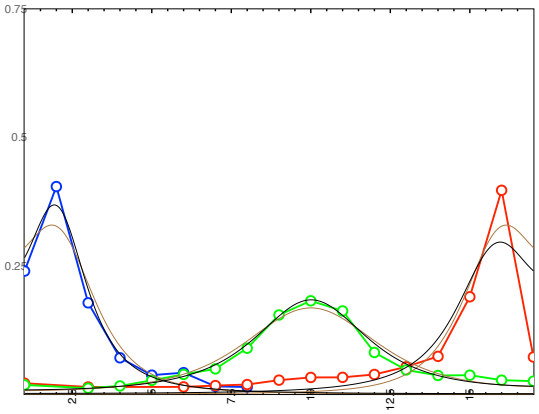

Figure 3: Finnish $I = J = 17$ and $i = 2, 10, 16$.

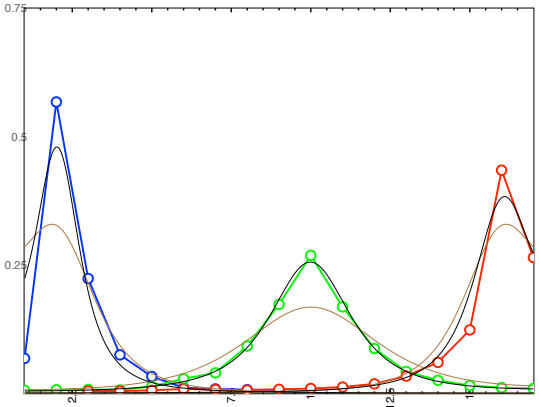

Figure 4: French $I = J = 17$ and $i = 2, 10, 16$.

show fitted and language-independent (in brown) reflected t distributions, respectively, with $r = 2$. The latter curves are mirror images for $i = 2$ and $i = 16$.

Figure 3 shows Finnish alignment probabilities. This is a great triumph for the language-independent parameters postulated a priori. We conclude that Finnish—a non-Indo-European language—is so different from English, that the latent alignment probabilities contain very little information. There is considerable correlation between words in the middle of Finnish sentences and words toward the end of English sentences, and vice versa. The situation is similar to that with German.

Figure 4 shows French alignment probabilities. The fit is extremely good and the variances are small. French-English was one of the first transla-

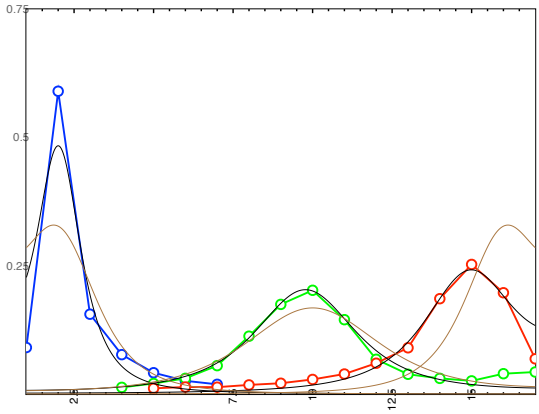

Figure 5: German $I = J = 17$ and $i = 2, 10, 16$.

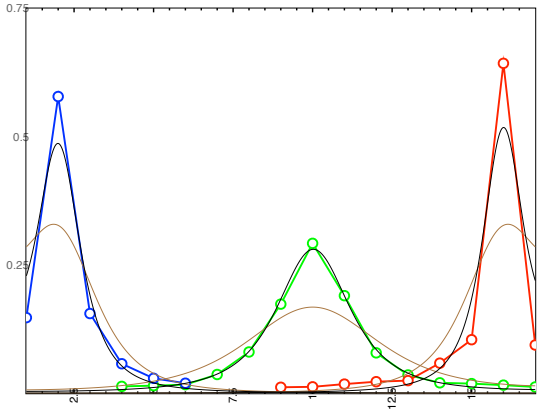

Figure 6: Swedish $I = J = 17$ and $i = 2, 10, 16$.

tion pairs to be attempted by statistical means, and one for which machine translation works the best.

Figure 5 shows German alignment probabilities. Note the probability increase towards the end for the data set $i = 10$, making this latent distribution multimodal, albeit just barely. For the data set $i = 16$, the language-independent mode is an entire point off, and the latent variance is much larger than for the data set $i = 2$. Contrast this with the situation for Swedish and French. We believe the stacking of verbs at the end of German clauses for these effects responsible to be.

Figure 6 shows Swedish alignment probabilities. The fit is extremely good and the variances are small. Swedish is one of the languages for which machine translation into English works the best.

The language-independent distributions are consistently too conservative—except for Finnish—indicating that our choice of language-independent variance is too high, erring on the side of caution.

5.3 Alignment Experiment 3

We extracted translation probabilities using the IBM scheme of Section 2 both with the latent distributions (the original method) and the parametric distributions using the fitted and language-independent parameters. We then compared the runtime performance using the experimental set-up of Section 7.

Average BLEU Scores*					
Lang	Lat	Fit	L-I	HAL	Koehn
Fin	11.5	11.4	11.9	13.6	21.8
Fra	22.5	22.3	22.6	23.9	30.0
Ger	18.0	17.9	18.0	18.4	25.3
Swe	20.5	20.3	20.7	21.9	30.2

* BLEU scores, see Section 7 and (Papineni et al., 2002).

The latent (Lat) and fitted parametric (Fit) distributions perform on par, as do the language-independent parameters (L-I). This insensitivity of translation quality to the exact form of the alignment probabilities supports the observations made in (Fraser and Marcu, 2007).

Column HAL quotes the results from Section 7, which uses the language-independent parameters and correlation-weighted translation probabilities, compare Section 3, *but no iteration*. Column Koehn quotes the (Koehn, 2005) results, which uses a phrase-based model and a stack decoder.

6 The HAL Scheme

In which we jettison the entire iterative procedure, offer an alternative scheme, and name it HAL.

Assembling everything, we arrive at a new scheme, which we call—for lack of a better name—HAL.

$$
\begin{aligned}
p(t \rightsquigarrow s) &\propto (D(t,s) - \lambda) \cdot \frac{C(t,s)}{C(t)} \\
p(t \rightsquigarrow \epsilon) &\propto p_0 \quad ; \quad p(\epsilon \rightsquigarrow s) \propto C(s) \\
C(t,s) &= \sum_k \sum_{i,j} p(i \mapsto j \mid i, I_k, J_k) \cdot \delta_{t,t_i^k} \cdot \delta_{s,s_j^k} \\
D(t,s) &= \frac{C(t,s) \cdot C()}{C(t) \cdot C(s)} \quad ; \quad C() = \sum_{t,s} C(t,s) \\
C(t) &= \sum_s C(t,s) \quad ; \quad C(s) = \sum_t C(t,s)
\end{aligned}
$$

with ($\hat{\imath} = i - \frac{1}{2}$ and $\hat{\jmath} = j - \frac{1}{2}$)

$$\begin{aligned} p(i \mapsto j \mid i, I, J) &= \\ &= \sum_{k=-\infty}^{\infty} f(\pm\hat{\jmath} + 2kJ; \mu_{\hat{\imath}IJ}, \sigma_{\hat{\imath}IJ}) \\ f(x; \mu, \sigma) &= \frac{1}{2\sqrt{2}\sigma} \cdot \left(1 + \frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2\right)^{-\frac{3}{2}} \\ \mu_{\hat{\imath}IJ} &= \frac{\hat{\imath}}{I} \cdot J \quad ; \quad \sigma^2_{\hat{\imath}IJ} = \frac{\hat{\imath}}{I} \cdot \frac{I - \hat{\imath}}{I} \cdot J \end{aligned}$$

Here λ is a real parameter. Good values include 0 and 1. Alternatively, one can use some other function of the modified Dice coefficient $D(t, s)$ in place of the factor $D(t, s) - \lambda$, for example $\ln D(t, s)$.

Note the absence of iteration: all right-hand-side quantities are known. This is a single-pass scheme.

The alignment probability estimates are without doubt needlessly complicated, and would benefit from simplification—we only need some smoothed window of roughly that shape. One candidate would be a binomial distribution $\mathrm{Bin}\left(J - 1, \frac{\hat{\imath}}{I}\right)$ shifted one position, were it not so close to a Gaussian.

7 Translation Experiments

In which we compare the IBM and HAL schemes.

We compared the IBM and HAL schemes on the EuroParl 6.0 parallel corpora (Koehn, 2005), translating from Finnish, French, German, and Swedish into English. All words were down-cased; unknown words were translated verbatim. The entire data sets were used to train the models, save a few thousand sentences at the very end, which were reserved for runtime testing.

Bilexical probabilities were extracted using the IBM scheme of Section 2 (half a dozen iterations) and the HAL scheme of Section 6 (a single pass), in both cases of model 2. The probabilities were symmetrized by taking the geometric mean with their converse probabilities and renormalizing

$$p'(t \rightsquigarrow s) \quad \propto \quad \sqrt{p(t \rightsquigarrow s) \cdot p(s \rightsquigarrow t)}$$

which is a standard procedure.

The word-based Viterbi decoder, similar to Pharaoh (Koehn, 2004a), allows word insertions, deletions, and inversions with heuristic penalties. It employs a powerful, interpolated word hexagram language model extracted from the training set.

One cannot expect the translation system to reproduce the reference translation exactly. We gauge translation quality by word n-gram BLEU scores (Papineni et al., 2002) for $n = 1 \ldots 4$, and, as is customary, the average BLEU score is the geometric mean of the n-gram scores with a multiplicative brevity penalty.

In a Bernoulli trial with probability p repeated N times, σ^2 is $\frac{p(1-p)}{N}$. As $\mathrm{erf}(1.386) = 0.95$, a difference of 1.386σ is 5% significant using a Gaussian approximation. For $p = 0.25$ and $N = 1\,000$, the threshold is 1.90 BLEU score points. According to (Koehn, 2004b), this is too conservative.

7.1 Translation Quality and Processing Speed

We translated the last 1 000 sentences of each language data set into English. In addition to the IBM and HAL schemes, we tested initializing the IBM model with the output from the HAL model (rows H+I). We also report the results from (Koehn, 2005) as a comparison.

BLEU Scores						
Lang	Model	Uni	Bi	Tri	Tet	Ave
Fin	IBM	45.9	19.8	9.0	4.2	11.5
	HAL	45.8	20.7	9.6	4.8	13.6
	H+I	48.1	21.5	10.0	4.8	13.1
	Koehn	—	—	—	—	**21.8**
Fra	IBM	49.3	28.9	17.1	10.5	22.5
	HAL	51.0	30.7	18.7	11.8	23.9
	H+I	51.0	30.4	18.4	11.4	23.8
	Koehn	—	—	—	—	**30.0**
Ger	IBM	47.6	24.7	13.0	7.2	18.0
	HAL	47.1	24.5	13.1	7.5	18.4
	H+I	48.6	25.5	13.7	7.8	18.7
	Koehn	—	—	—	—	**25.3**
Swe	IBM	50.2	28.1	15.9	9.6	20.5
	HAL	50.1	28.9	16.8	10.3	21.9
	H+I	51.4	29.5	17.0	10.4	21.6
	Koehn	—	—	—	—	**30.2**

Language	Size⋆	Model	Time (hh:mm)	
			Training	Testing
Finnish	1.74M	IBM	9:37	0:29
		HAL	3:31	0:39
		H+I	6:46†	0:23
French	1.82M	IBM	9:55	0:46
		HAL	2:39	0:58
		H+I	7:19†	0:30
German	1.73M	IBM	9:56	0:48
		HAL	2:52	0:48
		H+I	7:05†	0:30
Swedish	1.67M	IBM	8:23	0:26
		HAL	2:36	0:34
		H+I	6:09†	0:23

⋆ Size of training corpus in (mega) sentence pairs.
†Total time, actually.

Most importantly, *the HAL scheme outperforms the IBM scheme and it is much faster to train.* While the improvement is not quite statistically significantly for all language pairs, it most certainly is collectively. Applying the IBM scheme to the HAL bilexical probabilities as a post-processor (H+I) speeds up translation by one third, but more than doubles training times. Still, these two operations are faster than training the IBM model initialized with uniform distributions. Unfortunately, it degrades performance. A similar speed-up can however be obtained simply by using $\lambda = 1$ instead of $\lambda = 0$; this is just an effect of bilexicon size.

The HAL scheme is better at controlling the number of tag-along words, which make word insertions more effective. An example would be translating *Europaparlamentet* as *the European parliament*—the Swedish suffix *-et* indicates definiteness—which is possible with the bilexicon entries *the* $\rightsquigarrow$ *Europaparlamentet*, *European* $\rightsquigarrow$ *Europaparlamentet*, and *parliament* $\rightsquigarrow$ *Europaparlamentet*. A larger value of λ (see Section 3) yields fewer tag-along words.

Instead using $\lambda = 1$ or $g(D'(t,s)) = \ln D'(t,s)$ yielded similar but often slightly lower BLEU scores, yet still (almost) significantly higher than the IBM models. The function $g(D'(t,s)) = \ln D'(t,s)$ scored the best in other experiments on transliteration, where a tighter bilexicon was preferable.

7.2 Transitional Species

We tested both schemes with and without the Dice coefficient factor in the bilexical estimates, and the IBM scheme with fitted parametric (F) and latent (L) alignment probabilities on a subset of the Swedish/English data.

Swedish-to-English; 100 000 sentence training set							
Model			BLEU Scores				
I	A	D	Uni	Bi	Tri	Tet	Ave
Y	L	0	48.40	24.54	12.71	6.90	18.0
Y	F	0	49.23	25.11	12.86	6.93	18.2
Y	L	1	44.64	19.15	8.75	4.24	13.3
Y	F	1	48.50	23.28	11.48	5.92	16.6
N	F	0	45.38	22.50	11.60	6.39	16.6
N	F	1	49.32	26.00	13.93	7.94	19.4

I = Y/N with/without iteration.
A = L/F latent/parametric align. probs.
D = 1/0 with/without the Dice factor.
IBM is I=Y, A=E, D=0 (top entry).
HAL is I=N, A=F, D=1 (bottom entry).
The rest are transitional species.

The parametric distribution always improves translation quality. The Dice coefficient factor helps only with the parametric distribution, without iteration: it mixes poorly with the IBM weights $p_k(i \mid j)$. It is however essential when omitting iteration.

8 Summary and Conclusions

IBM schemes estimate bilexical probabilities by combining weighted coocurrence counts. These have the correct dimension, but do not measure word correlation. Using dimensional analysis, we found another combination—with an additional Dice coefficient factor—of the right dimension that does measure word correlation. It applies equally well to phrase counts as to word counts.

Our experiments indicate that alignment probabilities adopt Student's t distributions with predictable parameters, folded into discrete intervals. There is thus no need to determine alignment probabilities empirically. Continuous distributions are practical for phrase alignment probabilities.

Combining these observations, we removed the entire iterative procedure. The resulting scheme, named HAL, is applicable to all IBM models and to word-based and phrase-based approaches alike.

In our experiments, a word-based IBM model 2 was pitted against its HAL analog. The latter outperformed the former, and was much faster to train. We also found that, while the parametric distribution constitutes an improvement in general, the added

Dice coefficient factor in the bilexical probability estimates is needed to replace iteration, and at odds with the IBM iterative procedure.

The HAL scheme is widely applicable. Compared to the IBM scheme, it improves translation quality and eliminates iteration, thus radically reducing training times. The λ parameter allows controlling bilexicon size. This is convenient when using HAL, instead of IBM model 2, to provide a set of bilexical probabilities for seeding more elaborate models, or for extracting biphrasal probabilities or lexical features for other translation models.

Acknowledgements

This research was supported by the T4ME network of excellence (IST-249119), funded by the DG INFSO of the European Commission through the Seventh Framework Programme. It has benefited greatly from feedback from Bob Carpenter, Joakim Nivre, Jan Odijk, Khalil Sima'an, Atro Voutilainen, and the anonymous reviewers of ACL'12, AMTA'12, EACL'12, EAMT'12, and EMNLP'12.

References

Christopher M. Bishop. 2006. *Machine Learning and Pattern Recognition*. Springer Verlag.

Peter F. Brown, Stephen A. Della Pietra, Vincent J. Della Pietra, and R. L. Mercer. 1993. The mathematics of statistical machine translation: Parameter estimation. *Computational Linguistics*, 19:263–311.

L. R. Dice. 1945. Measures of the amount of ecologic association between species. *Journal of Ecology*, 26:297–302.

Alexander Fraser and Daniel Marcu. 2007. Measuring word alignment quality for statistical machine translation. *Comput. Linguist.*, 33:293–303, September.

Hans G. Hornung. 2006. *Dimensional Analysis*. Dover.

Philipp Koehn. 2004a. Pharaoh: a beam search decoder for phrase-based statistical machine translation models. In *Procs. AMTA*.

Philipp Koehn. 2004b. Statistical significance tests for machine translation evaluation. In *Procs. EMNLP*.

Philipp Koehn. 2005. Europarl: A parallel corpus for statistical machine translation. In *Procs. Machine Translation Summit X*, pages 79–86.

Franz Josef Och and Hermann Ney. 2003. A systematic comparison of various statistical alignment models. *Computational Linguistics*, 29.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Procs. ACL*, pages 311–318.

F. Smadja, K. R. McKeown, and V. Hatzivassiloglou. 1996. Translating collocations for bilingual lexicons: A statistical approach. *Computational Linguistics*, 22:1–38.

Nassim Nicholas Taleb. 2007. *The Black Swan : The Impact of the Highly Improbable*. Random House.

9 The Phrase-HAL Scheme

In which we extend the HAL scheme to phrases.

Let $\sigma = s_1 \dots s_J$ be a string of words, and let $\sigma_{i-1,j}$ refer to the substring consisting of the ith to jth word of σ, here $s_i \dots s_j$.

We now generalize the HAL scheme as follows.

$$
\begin{aligned}
p(\tau \rightsquigarrow \sigma) &\propto (D(\tau,\sigma) - \lambda) \cdot \frac{C(\tau,\sigma)}{C(\tau)} \\
p(\tau \rightsquigarrow \epsilon) &\propto p_0 \quad ; \quad p(\epsilon \rightsquigarrow \sigma) \propto C(\sigma) \\
C(\tau,\sigma) &= \sum_k \sum_{i=0}^{I_k-1} \sum_{i'=i+1}^{I_k} \sum_{j=0}^{J_k-1} \sum_{j'=j+1}^{J_k} \delta_{\tau,\tau^k_{i,i'}} \cdot \delta_{\sigma,\sigma^k_{j,j'}} \cdot \\
&\qquad \cdot p(\{i,i'\} \mapsto \{j,j'\} \mid \{i,i'\}, I_k, J_k) \\
D(\tau,\sigma) &= \frac{C(\tau,\sigma) \cdot C()}{C(\tau) \cdot C(\sigma)} \quad ; \quad C() = \sum_{\tau,\sigma} C(\tau,\sigma) \\
C(\tau) &= \sum_\sigma C(\tau,\sigma) \quad ; \quad C(\sigma) = \sum_\tau C(\tau,\sigma)
\end{aligned}
$$

Again λ is a real parameter, such as 0 or 1.

We could, for example, decompose the alignment probabilities thus.

$$
\begin{aligned}
p(\{i,i'\} \mapsto \{j,j'\} \mid \{i,i'\}, I, J) &= \\
&= q(j'-j \mid i'-i) \cdot p\left(\hat{\imath} \mapsto \hat{\jmath} \mid \hat{\imath}, I, J\right)
\end{aligned}
$$

Now $\hat{\imath} = \frac{i+i'}{2}$ and $\hat{\jmath} = \frac{j+j'}{2}$. Note the difference from the word-based case: $-\frac{1}{2}$ is built in. The probabilities $q(l_j \mid l_i)$ model phrase length correspondence and remain to be specified. We again have

$$
p(\hat{\imath} \mapsto \hat{\jmath} \mid \hat{\imath}, I, J) = \sum_{k=-\infty}^{\infty} f(\pm\hat{\jmath} + 2kJ; \mu_{\hat{\imath}IJ}, \sigma_{\hat{\imath}IJ})
$$

and (here, σ is something completely different)

$$
\begin{aligned}
f(x;\mu,\sigma) &= \frac{1}{2\sqrt{2}\sigma} \cdot \left(1 + \frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2\right)^{-\frac{3}{2}} \\
\mu_{\hat{\imath}IJ} &= \frac{\hat{\imath}}{I} \cdot J \quad ; \quad \sigma^2_{\hat{\imath}IJ} = \frac{\hat{\imath}}{I} \cdot \frac{I-\hat{\imath}}{I} \cdot J
\end{aligned}
$$