

A Systematic Bayesian Treatment of the IBM Alignment Models

Yarin Gal

Department of Engineering
University of Cambridge

Cambridge, CB2 1PZ, United Kingdom
yg279@cam.ac.uk

Phil Blunsom

Department of Computer Science
University of Oxford

Oxford, OX1 3QD, United Kingdom
Phil.Blunsom@cs.ox.ac.uk

Abstract

The dominant yet ageing IBM and HMM word alignment models underpin most popular Statistical Machine Translation implementations in use today. Though beset by the limitations of implausible independence assumptions, intractable optimisation problems, and an excess of tunable parameters, these models provide a scalable and reliable starting point for inducing translation systems. In this paper we build upon this venerable base by recasting these models in the non-parametric Bayesian framework. By replacing the categorical distributions at their core with hierarchical Pitman-Yor processes, and through the use of collapsed Gibbs sampling, we provide a more flexible formulation and sidestep the original heuristic optimisation techniques. The resulting models are highly extendible, naturally permitting the introduction of phrasal dependencies. We present extensive experimental results showing improvements in both AER and BLEU when benchmarked against Giza++, including significant improvements over IBM model 4.

1 Introduction

The IBM and HMM word alignment models (Brown et al., 1993; Vogel et al., 1996) have underpinned the majority of statistical machine translation systems for almost twenty years. The key attraction of these models is their principled probabilistic formulation, and the existence of (mostly) tractable algorithms for their training.

The dominant Giza++ implementation of the IBM models (Och and Ney, 2003) employs a variety of exact and approximate EM algorithms to optimise categorical alignment distributions. While effective, this parametric approach results in a significant number of parameters to be tuned and intractable summations over the space of alignments for models 3 and 4. Giza++ hides the hyperparameters with defaults and approximates the intractable expectations using restricted alignment neighbourhoods. However this approach was shown to often return alignments with probabilities well below the true maxima (Ravi and Knight, 2010).

To overcome perceived limitations with the word based and non-syntactic nature of the IBM models many alternative approaches to word alignment have been proposed (e.g. (DeNero et al., 2008; Cohn and Blunsom, 2009; Levenberg et al., 2012)). While interesting results have been reported, these alternatives have failed to dislodge the IBM approach.

In this paper we proposed to retain the original generative stories of the IBM models, while replacing the inflexible categorical distributions with hierarchical Pitman-Yor (PY) processes – a mathematical tool which has been successfully applied to a range of language tasks (Teh, 2006b; Goldwater et al., 2006; Blunsom and Cohn, 2011). In the context of language modelling, the hierarchical PY process was shown to roughly correspond to interpolated Kneser-Ney (Kneser and Ney, 1995; Teh, 2006a). The key contribution of the hierarchical PY formulation is that it provides a principle probabilistic framework that easily extends to latent variable models, such as those used

for alignment, for which a Kneser-Ney formulation is unclear. While Bayesian priors have previously been applied to IBM model 1 (Riley and Gildea, 2012), in this work we go considerably further by implementing non-parametric priors for the full Giza++ training pipeline.

Inference for the proposed models and their hyper-parameters is done with Gibbs sampling. This eliminates the intractable summations over alignments and the need for tuning hyper-parameters. Further, we exploit the highly extendible nature of the hierarchical PY process to implement improvements to the original models such as the introduction of phrasal dependencies.

We present extensive experimental results showing improvements in both BLEU scores and AER when compared to Giza++. The demonstrated improvements over IBM model 4 suggest that the heuristics used in the implementation of the EM algorithm for this model were suboptimal.

We begin with a formal presentation of the hierarchical PY process used in our Bayesian approach to replace the original categorical distributions. Section 3 introduces our Bayesian formulation of the word alignment models, while its inference scheme is presented in the following section. Finally, the experimental results evaluating our models against the originals are given in section 5, demonstrating the superiority of the non-parametric approach.

2 The hierarchical PY process

Before giving the formal definition for the hierarchical Pitman-Yor (PY) process, we first give some intuition into how this distribution works and why it is commonly used to model problems in natural language.

The hierarchical PY process is an atomic distribution that can share its atoms between different levels in a hierarchy. When used for language modelling it captures the probability of observing a word after any given sequence of n words. It does so by interpolating the observed frequency of the whole sequence followed by the word of interest, with the observed frequency of a shorter sequence followed by the word of interest. This interpolation is done in such a way that tokens in a more specific distribution are interpolated with types in a less specific one.

If there is sufficient evidence for the whole word sequence, i.e. it is not sparse in the corpus, higher weight will be given to the frequency of the word of interest after the more specific sequence than the shorter one. If the sequence was not observed frequently and there is not enough information to model whether the word of interest follows after it frequently or not, the process will back-off to the shorter sequence and assign higher weight to its frequency instead. This is done in a recursive fashion, decreasing the sequence length by one every time until the probability is interpolated with the uniform distribution, much like interpolated Kneser-Ney, the state of the art for language modelling.

Unlike Kneser-Ney, the hierarchical PY approach naturally extends to model complicated conditional distributions involving latent variables. Moreover, almost all instances of priors with categorical distributions can be replaced by the PY process, where in its most basic representation (with no conditional) it provides a flexible model of power law frequencies.

The PY process generalises a number of simpler distributions. The Dirichlet distribution is a distribution over discrete probability mass functions of a certain given length which is often used to model prior beliefs on parameter sparsity in machine learning problems. The Dirichlet process generalises the Dirichlet distribution to a distribution over infinite sequences of non-negative reals that sum to one and is often used for nonparametric Bayesian inference. The PY process is used in the context of natural language processing as it further generalises the Dirichlet process by adding an additional degree of freedom that enables it to produce power-law discrete probability mass functions that resemble those seen experimentally in corpora (Goldwater et al., 2006).

Formally, draws from the PY process $G_1 \sim PY(d, \theta, G_0)$ with a discount parameter $0 \leq d < 1$, a strength parameter $\theta > -d$, and a base distribution G_0 , are constructed using a Chinese restaurant process analogy as follows:

$$X_{n+1}|X_1, \dots, X_n \sim \sum_{k=1}^K \frac{m_k - d}{\theta + n} \delta_{y_k} + \frac{\theta + dK}{\theta + n} G_0$$

Where m_k denotes the number of X_i s (customers) assigned to y_k (a table) and K is the total number of y_k s drawn from G_0 .

Hierarchical PY processes (Teh, 2006b), PY processes where the base distribution is itself a PY process, were developed as an extension which is often used in the context of natural language processing due to their relationship to back-off smoothing. Denoting a context of atoms \mathbf{u} as $(w_{i-l}, \dots, w_{i-1})$, the hierarchical PY process is defined using the above definition of the PY process by:

$$\begin{aligned} w_i &\sim G_{\mathbf{u}} \\ G_{\mathbf{u}} &\sim PY(d_{|\mathbf{u}|}, \theta_{|\mathbf{u}|}, G_{\pi(\mathbf{u})}) \\ &\dots \\ G_{(w_{i-1})} &\sim PY(d_1, \theta_1, G_{\emptyset}) \\ G_{\emptyset} &\sim PY(d_0, \theta_0, G_0) \end{aligned}$$

where $\pi(\mathbf{u}) = (w_{i-l+1}, \dots, w_{i-1})$ is the suffix of \mathbf{u} , $|\mathbf{u}|$ denotes the length of context \mathbf{u} , and G_0 is a base distribution.

3 A Bayesian approach to word alignment

In this work we replace the categorical distributions at the heart of statistical alignment models with PY processes. We start by describing the revised models for IBM model 1 and the HMM alignment model, before continuing to the more advanced IBM models 3 and 4. Throughout this section, we assume that the base distributions in our models (denoted G_0, H_0 , etc.) are uniform over all atoms, and omit the strength and concentration parameters of the PY process for clarity. We use subscripts to denote the hierarchy, and lower-case superscripts to denote a fixed condition (for example, G_0^m is the (uniform) base distribution that is determined uniquely for each possible foreign sentence length m).

3.1 Model 1 and the HMM alignment model

The most basic word alignment model, IBM model 1, can be described using the following generative process (Brown et al., 1993): Given an English sentence $E = e_1, \dots, e_l$, first choose a length m for the foreign sentence F . Next, choose a vector of random word positions from the English sentence $A = a_1, \dots, a_m$ to be the alignment, and then for each foreign word f_i choose a translation from the English word e_{a_i} aligned to it by A . The existence of a NULL word at the beginning of the English sentence is assumed, a word to which spurious words in

the foreign sentence can align. From this generative process the following probability model is derived:

$$P(F, A | E) = p(m|l) \times \prod_{i=1}^m p(a_i) p(f_i | e_{a_i})$$

Where $p(a_i) = \frac{1}{l+1}$ is uniform over all alignments and $p(f_i | e_{a_i}) \sim \text{Categorical}$.

In our approach we model these distributions using hierarchical PY processes instead of the categorical distributions. Thus we place the following assumptions on IBM model 1:

$$\begin{aligned} a_i | m &\sim G_0^m \\ f_i | e_{a_i} &\sim H_{e_{a_i}} \\ H_{e_{a_i}} &\sim PY(H_{\emptyset}) \\ H_{\emptyset} &\sim PY(H_0) \end{aligned}$$

In this probability modelling we assume that the alignment positions are determined using the uniform distribution, and that word translations are generated depending on the source word – the probability of translating to a specific foreign word depends on the observed frequency of pairs of the foreign word and the given source word. We back-off to the frequencies of the foreign words when the source word wasn't observed frequently.

The HMM alignment model uses the Hidden Markov Model to find word alignments. It treats the translations of the words of the English sentence as observables and the alignment positions as the latent variables to be discovered. Its generative process can be described in an abstract way as follows: we determine the length of the foreign sentence and then iterate over the words of the source sentence emitting translations for each word to fill-in the words in the foreign sentence from left to right.

The following probability model is derived for the HMM alignment model (Vogel et al., 1996):

$$\begin{aligned} P(F, A | E) &= \\ p(m|l) \times \prod_{i=1}^m &p(a_i | a_{i-1}, m) \times p(f_i | e_{a_i}) \end{aligned}$$

For the HMM alignment model we replace the categorical translation distribution $p(f_i | e_{a_i})$ with the same one we used for model 1, and

replace the categorical distribution for the transition $p(a_i|a_{i-1}, m)$ with a hierarchical PY process with a longer sequence of alignment positions in the conditional:

$$\begin{aligned} a_i|a_{i-1}, m &\sim G_{a_{i-1}}^m \\ G_{a_{i-1}}^m &\sim PY(G_\emptyset^m) \\ G_\emptyset^m &\sim PY(G_0^m) \end{aligned}$$

We use a unique distribution for each foreign sentence length, and condition the position on the previous alignment position, backing-off to the HMM's stationary distribution over alignment positions.

3.2 Models 3 and 4

IBM models 3 and 4 introduce the concept of a word's fertility, the number of foreign words that are generated from a specific English word. These models can be described using the following generative process. Given an English sentence E , first determine the length of the foreign sentence: for each word in the English sentence e_i choose a fertility, denoted ϕ_i . Every time a word is generated, an additional spurious word from the NULL word in the English sentence can be generated with a fixed probability. After the foreign sentence length is determined translate each English word into its foreign equivalent (including the NULL word) in the same way as for model 1. Finally, non-spurious words are rearranged into the final word positions and the spurious words inserted into the empty positions. In model 3 this is done with a zero order HMM, and in model 4 with two first order HMMs. One controls the distortion of the head of each English word (the first foreign word generated from it) relative to the centre (denoted here \odot) of the set of foreign words generated from the previous English word, and the other controls the distortion within the set itself by conditioning the word position on the previous word position.

For the probability model, we follow the notation of Och and Ney (2003) and define the alignment as a function from the source sentence positions i to $B_i \subset \{1, \dots, m\}$ where the B_i 's form a partition of the set $\{1, \dots, m\}$. The fertility of the English word i is $\phi_i = |B_i|$, and we use $B_{i,k}$ to refer to the k th element of B_i in ascending order.

Using the above notation, the following probability model is derived (Och and Ney, 2003):

$$\begin{aligned} P(F, A|E) = &p(B_0|B_1, \dots, B_l) \times \prod_{i=1}^l p(B_i|B_{i-1}, e_i) \\ &\times \prod_{i=0}^l \prod_{j \in B_i} p(f_j|e_i) \end{aligned}$$

For model 3 the dependence on previous alignment sets is ignored and the probability $p(B_i|B_{i-1}, e_i)$ is modelled as

$$p(B_i|B_{i-1}, e_i) = p(\phi_i|e_i) \phi_i! \prod_{j \in B_i} p(j|i, m),$$

whereas in model 4 it is modelled using two HMMs:

$$\begin{aligned} p(B_i|B_{i-1}, e_i) = &p(\phi_i|e_i) \times p_{=1}(B_{i,1} - \odot(B_{i-1})|\cdot) \\ &\times \prod_{k=2}^{\phi_i} p_{>1}(B_{i,k} - B_{i,k-1}|\cdot) \end{aligned}$$

For both these models the spurious word generation is controlled by a binomial distribution:

$$p(B_0|B_1, \dots, B_l) = \binom{m - \phi_0}{\phi_0} (1 - p_0)^{m - 2\phi_0} p_1^{\phi_0} \frac{1}{\phi_0!}$$

for some parameters p_0 and p_1 .

Replacing the categorical priors with hierarchical PY process ones, we set the translation and fertility probabilities $p(\phi_i|e_i) \prod_{j \in B_i} p(f_j|e_i)$ using a common prior that generates translation sequences:

$$\begin{aligned} (f^1, \dots, f^{\phi_i})|e_i &\sim H_{e_i} \\ H_{e_i} &\sim PY(H_{e_i}^{FT}) \\ H_{e_i}^{FT}((f^1, \dots, f^{\phi_i})) &= H_{e_i}^F(\phi_i) \prod_j H_{(f^{j-1}, e_i)}^T(f^j) \\ H_{e_i}^F &\sim PY(H_\emptyset^F) & H_{(f^{j-1}, e_i)}^T &\sim PY(H_{e_i}^T) \\ H_\emptyset^F &\sim PY(H_0^F) & H_{e_i}^T &\sim PY(H_\emptyset^T) \\ H_\emptyset^T &\sim PY(H_0^T) \end{aligned}$$

Here we used superscripts for the indexing of words which do not have to occur sequentially in the sentence. We generate sequences instead of individual words and fertilities, and fall-back onto these only in sparse cases. For example, when aligning the English sentence “I don't speak French” to its

French translation “Je ne parle pas français”, the word “not” will generate the phrase (“ne”, “pas”), which will later on be distorted into its place around the verb.

The distortion probability for model 3, $p(j|i, m)$, is modelled simply as depending on the position of the source word i and its class:

$$\begin{aligned} j|(C(e_i), i), m &\sim G_{(C(e_i), i)}^m \\ G_{(C(e_i), i)}^m &\sim PY(G_i^m) \\ G_i^m &\sim PY(G_\emptyset^m) \\ G_\emptyset^m &\sim PY(G_0^m) \end{aligned}$$

where we back-off to the source word position and then to the frequencies of the alignment positions.

As opposed to this simple mechanism, in the distortion probability for IBM model 4 there exist two distinct probability distributions. The first probability distribution $p_{=1}$ controls the head distortion:

$$\begin{aligned} B_{i,1} - \odot(B_{i-1}) | (C(e_i), C(f_{B_{i,1}})), m \\ &\sim G_{(C(e_i), C(f_{B_{i,1}}))}^m \\ G_{(C(e_i), C(f_{B_{i,1}}))}^m &\sim PY(G_{C(f_{B_{i,1}})}^m) \\ G_{C(f_{B_{i,1}})}^m &\sim PY(G_\emptyset^m) \\ G_\emptyset^m &\sim PY(G_0^m) \end{aligned}$$

In this probability modelling we model the jump size itself, as depending on the word class for the source word and the word class for the proposed foreign word, backing-off to the proposed foreign word class and then to the relative jump frequencies.

The second probability distribution $p_{>1}$ controls the distortion within the set of words:

$$\begin{aligned} B_{i,j} - B_{i,j-1} | C(f_{B_{i,j}}), m &\sim H_{C(f_{B_{i,j}})}^m \\ H_{C(f_{B_{i,j}})}^m &\sim PY(H_\emptyset^m) \\ H_\emptyset^m &\sim PY(H_0^m) \end{aligned}$$

Here we again model the jump size as depending on the word class for the proposed foreign word, backing-off to the relative jump frequencies.

Lastly, we add to this probability model a treatment for fertility and translation of NULL words. The fertility generation follows the idea of the original model, where the number of spurious words is

determined by a binomial distribution created from a set of Bernoulli experiments, each one performed after the translation of a non-spurious word. We use an indicator function I to signal whether a spurious word was generated after a non-spurious word ($I = 1$) or not ($I = 0$).

$$\begin{aligned} I = 0, 1 | l &\sim H_l^{NF} \\ H_l^{NF} &\sim PY(H_\emptyset^{NF}) \\ H_\emptyset^{NF} &\sim PY(H_0^{NF}) \end{aligned}$$

Then, the translation of spurious words is done in a straightforward manner:

$$\begin{aligned} f_i &\sim H_\emptyset^{NT} \\ H_\emptyset^{NT} &\sim PY(H_0^{NT}) \end{aligned}$$

4 Inference

The Gibbs sampling inference scheme together with the Chinese Restaurant Franchise process (Teh and Jordan, 2009) are used to induce alignments for a parallel corpus. A set of restaurants \mathcal{S} is constructed and initialised either randomly or through a pipeline of alignment results from simpler models, and then at each iteration each alignment position is removed from the restaurants and re-sampled, conditioned on the rest of the alignment positions.

Denoting $\mathbf{e}, \mathbf{f}, \mathbf{a}$ the sets of all source sentences, their translations, and their corresponding alignments in our corpus, and denoting E, F, A a specific source sentence, its translation, and their corresponding alignment, where e_i is the i 'th word of the source sentence and f_j, a_j are the j 'th word in the foreign sentence and its alignment into the source sentence, we sample a new value for a_j using the univariate conditional distribution:

$$\begin{aligned} P(a_j = i | E, F, A_{-j}, \mathbf{e}_{-E}, \mathbf{f}_{-F}, \mathbf{a}_{-A}, \mathcal{S}_{-a_j}) \\ \propto P(F, (A_{-j}, a_j = i) | E, \mathbf{e}_{-E}, \mathbf{f}_{-F}, \mathbf{a}_{-A}, \mathcal{S}_{-a_j}) \end{aligned}$$

Where a minus sign in the subscript denotes the structure without the mentioned element, and \mathcal{S}_{-a_j} denotes the configuration of the restaurants after removing the alignment a_j .

This univariate conditional distribution is proportional to the probability assigned by the different models to an alignment sequence, where the restaurants replace the counts of the alignment positions

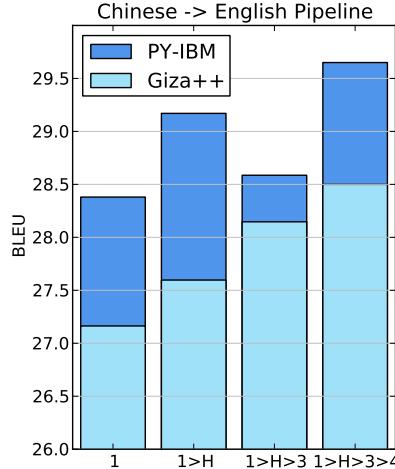


Figure 1: BLEU scores of pipelined Giza++ and pipelined PY-IBM translating from Chinese into English

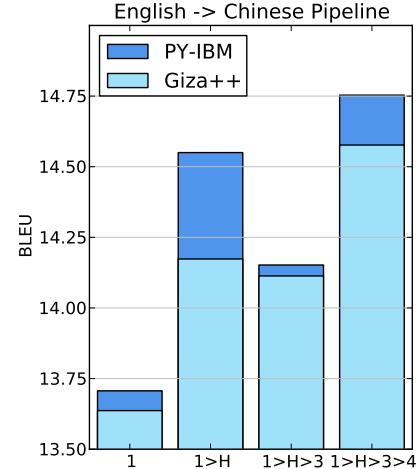


Figure 2: BLEU scores of pipelined Giza++ and pipelined PY-IBM translating from English into Chinese

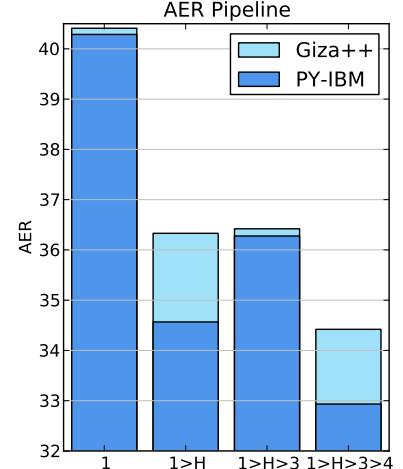


Figure 3: AER of pipelined Giza++ and pipelined PY-IBM aligning Chinese and English

in the prior. Maximum marginal decoding (Johnson and Goldwater, 2009) can then be used to get the MAP estimate of the probability distributions over the alignment positions for each sentence from the samples extracted from the Gibbs sampler.

In addition to sampling the alignments, we also place a uniform Beta prior on the discount parameters and a vague Gamma prior on the strength parameters, and sample them using slice sampling (Neal, 2003). The end result is that the alignment models have no free parameters to tune.

5 Experimental results

In order to assess our PY process alignment models (referred to as PY-IBM henceforth) several experiments were carried out to benchmark them against the original models (as implemented in Giza++). We evaluated the BLEU scores (Papineni et al., 2002) of translations from Chinese into English and from English into Chinese, as well as the alignment error rates (AER) of the induced symmetrised alignments compared to a human aligned dataset. Moses (Koehn et al., 2007) was used for the training of the SMT system and the symmetrisation (using the grow-diag-final procedure), with MERT (Och, 2003) used for tuning of the weights, and SRILM (Stolcke, 2002) to build the language model (5-grams based). The corpus used for training and evaluation was the Chinese

FBIS corpus. MT02 was used for tuning, and MT03 was used for evaluation. In each case we used one reference sentence in Chinese and 4 reference sentences in English.

Most translation systems rely on the Giza++ package in which the implementation of the original models is done by combining them in a pipeline. Model 1 and the HMM alignment model are run sequentially each for 5 iterations; then models 3 and 4 are run sequentially for 3 iterations each. This follows the observation of Och and Ney (2003) that bootstrapping from previous results assists the fertility algorithms find the best alignment neighbourhood in order to estimate the expectations.

We assessed the proposed models against the original models in a pipeline experiment where both systems were trained on a corpus starting at model 1, and used the results of the previous run to initialise the next one – noting the BLEU scores and AER at each step. The Gibbs samplers for the pipelined PY-IBM models were run for 50 iterations for each model and started accumulating samples after a burn-in period of 10 iterations, each experiment was repeated three times and the results averaged. As can be seen in figures 1 to 3, the pipelined PY-IBM models achieved higher BLEU scores across all steps, with the highest improvement of 1.6 percentage points in the pipelined HMM alignment models when translating

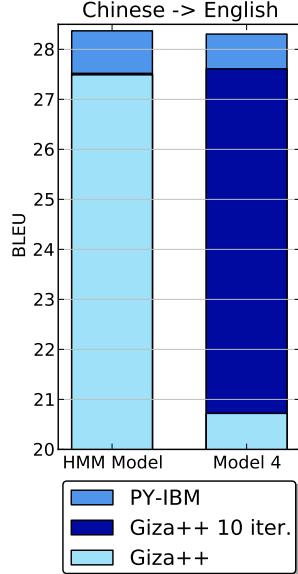


Figure 4: BLEU scores of Giza++’s and PY-IBM’s HMM model and model 4 translating from Chinese into English

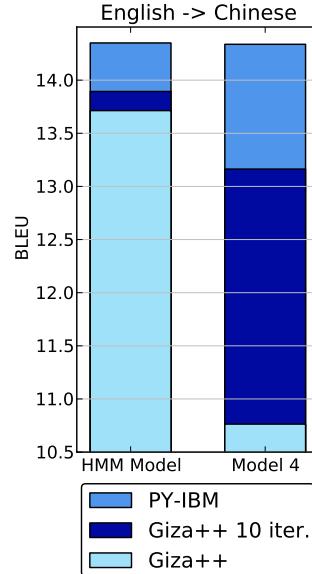


Figure 5: BLEU scores of Giza++’s and PY-IBM’s HMM model and model 4 translating from English into Chinese

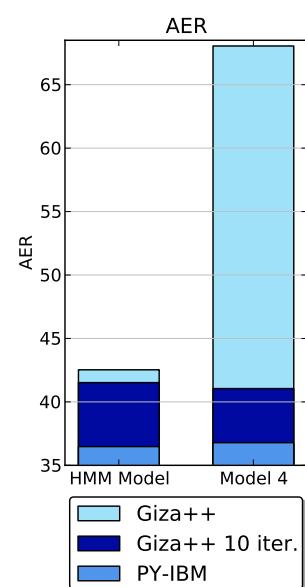


Figure 6: AER of Giza++’s and PY-IBM’s HMM model and model 4 aligning Chinese and English

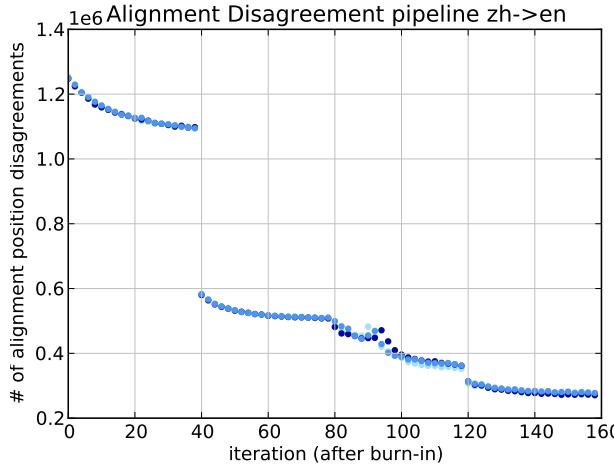


Figure 7: Alignment disagreement of the Chinese to English pipelined PY-IBM models for the 3 repetitions

from Chinese into English, and an improvement of 1.2 percentage points in the overall results after finishing the pipeline.

We also saw an improvement in AER for all models, where the pipelined PY-IBM model 4 achieved an error rate of 32.9, as opposed to the result obtained by the Giza++ pipelined model 4 of 34.4. We note an interesting observation that both Giza++ and PY-IBM model 3 underperformed compared to the previously run HMM alignment

model, as seen in the English to Chinese pipeline results and the AER pipeline results.

The alignment disagreement (the number of changed alignment positions between subsequent iterations) of the Chinese to English pipelined PY-IBM models (1 to 4) can be seen in fig. 7. This graph shows that each model in the pipeline reaches an alignment disagreement equilibrium after about 20 iterations, and that earlier models have greater initial deviation from their equilibrium than later models – which have an overall lower disagreement.

In order to assess the dependence of the fertility based models on the initialisation step another set of experiments was carried out. The models were trained with a randomly initialised set of alignments and assessed after a set number of iterations for the Giza++ models (5 and 10 for the Giza++ HMM alignment model, and 3 and 10 for the Giza++ IBM model 4), or after 100 iterations with a burn-in period of 10 iterations for the PY-IBM ones (we report the average of three runs for both models).

The results, reported in figures 4 to 6, show again that the PY-IBM model outperformed the Giza++ implementations, and to a large extent in the case of IBM model 4. This provides further evidence that the supposition underlying the neighbourhood

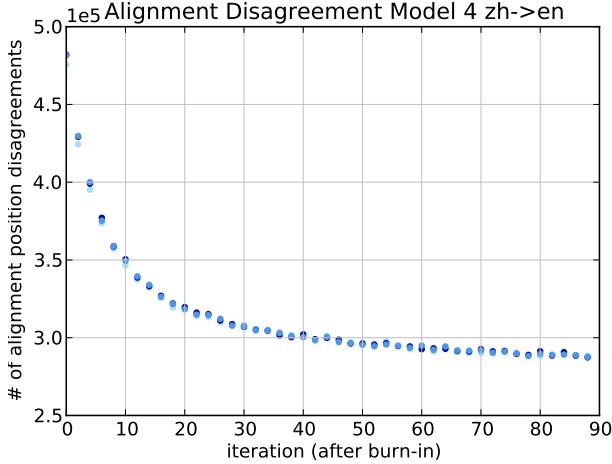


Figure 8: Alignment disagreement of the Chinese to English PY-IBM model 4 for the 3 repetitions

approximation for training models 3 and 4 – that there exists a small set of alignments on which most of the probability mass concentrates – is poor. An interesting observation to note is that the BLEU score of the non-pipelined PY-IBM model 4 is the same as the PY-IBM HMM model translating in both directions, as opposed to an improvement in the pipelined case. This suggests that the sampler might not have fully converged after 100 iterations for model 4 (the number of alignment disagreements for this experiment can be seen in figure 8). Further confirmation for this comes from the higher standard deviation of 0.54 observed for the PY-IBM model 4, as opposed to a standard deviation for the PY-IBM HMM model of 0.21 (which is still more significant than that of the pipelined PY-IBM model 4, whose standard deviation was 0.13).

Both the PY-IBM and the Giza++ trained models run in a linear time in the number of sentences, where due to the nature of MCMC sampling techniques, more iterations are required for its convergence. In our experiments, the running time of the unoptimised Gibbs sampler was 50 times slower than the optimised EM.

6 Discussion

The models described in this paper allow one to use non-parametric approaches to flexibly model word alignment distributions, overcoming a number of limitations of the EM algorithm for the fertility based alignment models. The models achieved a significant improvement in BLEU scores and AER

on the tested corpus, and are easy to extend without the need for additional modelling tools.

The alignment models proposed mostly follow the original generative stories while introducing additional phrasal conditioning into models 3 and 4. However there are many other areas in which we could make use of hierarchical tools to introduce new dependencies easily without running into sparsity problems.

One example is the extension of the transition history used in the HMM alignment model: IBM model 1 uses a uniform distribution over transitions, model 2 conditions on relative sentence positions, and the HMM model uses a first order dependency. One extension would be to use longer histories of n previous positions, handling sparsity with back-off.

Previously proposed approaches to extend the HMM alignment model include Och and Ney (2003)’s use of word classes and smoothing, and the combination of part-of-speech information of the words surrounding the source word (Brunner et al., 2009). Using our hierarchical model one could easily introduce such dependencies on the context words of the word to be translated and their part-of-speech information. This could assist in both translation and reordering disambiguation, and would incorporate back-off by using smaller and smaller contexts when such information is sparse.

Further improvements to models 3 and 4 could be carried out by introducing longer dependencies in the fertility and distortion distributions. Instead of conditioning on the previous word, one could use further information such as PoS tags, previously translated words, or previous fertilities. Additional research would involve the use of more effective variational inference algorithms for hierarchical PY process based models.

The PY-IBM models described in this paper were implemented within the Giza++ code base, and are available as an open source package for further development and research.¹

References

- Phil Blunsom and Trevor Cohn. 2011. A hierarchical Pitman-Yor process HMM for unsupervised part of

¹Available at github.com/yaringal/Giza-sharp

- speech induction. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 865–874, Portland, Oregon, USA, June. Association for Computational Linguistics.
- Peter F. Brown, Stephen A. Della Pietra, Vincent J. Della Pietra, and Robert L. Mercer. 1993. The mathematics of statistical machine translation: Parameter estimation. *Computational Linguistics*, 19(2):263–311.
- Jamie Brunning, Adrià de Gispert, and William Byrne. 2009. Context-dependent alignment models for statistical machine translation. In *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, NAACL '09, pages 110–118.
- Trevor Cohn and Phil Blunsom. 2009. A Bayesian model of syntax-directed tree to string grammar induction. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing*, pages 352–361, Singapore, August. Association for Computational Linguistics.
- John DeNero, Alexandre Bouchard-Côté, and Dan Klein. 2008. Sampling alignment structure under a Bayesian translation model. In *Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing*, pages 314–323, Honolulu, Hawaii, October. Association for Computational Linguistics.
- Sharon Goldwater, Tom Griffiths, and Mark Johnson. 2006. Interpolating between types and tokens by estimating power-law generators. In Y. Weiss, B. Schölkopf, and J. Platt, editors, *Advances in Neural Information Processing Systems 18*, pages 459–466. MIT Press, Cambridge, MA.
- Mark Johnson and Sharon Goldwater. 2009. Improving nonparametric Bayesian inference: experiments on unsupervised word segmentation with adaptor grammars. In *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, NAACL '09, pages 317–325.
- Reinhard Kneser and Hermann Ney. 1995. Improved backing-off for m-gram language modeling. *Acoustics, Speech, and Signal Processing, IEEE International Conference on*, 1:181–184.
- Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, Chris Dyer, Ondrej Bojar, Alexandra Constantin, and Evan Herbst. 2007. Moses: Open source toolkit for statistical machine translation. In *Proc. of the 45th Annual Meeting of the ACL (ACL-2007)*, Prague.
- Abby Levenberg, Chris Dyer, and Phil Blunsom. 2012. A Bayesian model for learning SCFGs with discontiguous rules. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 223–232, Jeju Island, Korea, July. Association for Computational Linguistics.
- Radford Neal. 2003. Slice sampling. *Annals of Statistics*, 31:705–767.
- Franz Josef Och and Hermann Ney. 2003. A systematic comparison of various statistical alignment models. *Computational Linguistics*, 29(1):19–52.
- Franz Josef Och. 2003. Minimum error rate training in statistical machine translation. In *Proc. of the 41st Annual Meeting of the ACL (ACL-2003)*, pages 160–167, Sapporo, Japan.
- Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proc. of the 40th Annual Meeting of the ACL and 3rd Annual Meeting of the NAACL (ACL-2002)*, pages 311–318, Philadelphia, Pennsylvania.
- Sujith Ravi and Kevin Knight. 2010. Does Giza++ make search errors? *Computational Linguistics*, 36(3):295–302, September.
- Darcey Riley and Daniel Gildea. 2012. Improving the ibm alignment models using variational bayes. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Short Papers - Volume 2*, ACL '12, pages 306–310.
- A. Stolcke. 2002. SRILM – an extensible language modeling toolkit. In *Proc. of the International Conference on Spoken Language Processing*.
- Y. W. Teh and M. I. Jordan. 2009. *Hierarchical Bayesian Nonparametric Models with Applications*. Cambridge University Press.
- Yee Whye Teh. 2006a. A Bayesian interpretation of interpolated Kneser-Ney. Technical report, National University of Singapore School of Computing.
- Yee Whye Teh. 2006b. A hierarchical bayesian language model based on pitman-yor processes. In *Proceedings of the 21st International Conference on Computational Linguistics and the 44th annual meeting of the Association for Computational Linguistics*, ACL-44, pages 985–992, Morristown, NJ, USA. Association for Computational Linguistics.
- Stephen Vogel, Hermann Ney, and Christoph Tillmann. 1996. HMM-based word alignment in statistical translation. In *Proc. of the 16th International Conference on Computational Linguistics (COLING '96)*, pages 836–841, Copenhagen, Denmark, August.