A Markov Model of Machine Translation using Non-parametric Bayesian Inference

Yang Feng and **Trevor Cohn**
Department of Computer Science
The University of Sheffield
Sheffield, United Kingdom
yangfeng145@gmail.com and t.cohn@sheffield.ac.uk

Abstract

Most modern machine translation systems use phrase pairs as translation units, allowing for accurate modelling of phrase-internal translation and reordering. However phrase-based approaches are much less able to model sentence level effects between different phrase-pairs. We propose a new model to address this imbalance, based on a word-based Markov model of translation which generates target translations left-to-right. Our model encodes word and phrase level phenomena by conditioning translation decisions on previous decisions and uses a hierarchical Pitman-Yor Process prior to provide dynamic adaptive smoothing. This mechanism implicitly supports not only traditional phrase pairs, but also gapping phrases which are non-consecutive in the source. Our experiments on Chinese to English and Arabic to English translation show consistent improvements over competitive baselines, of up to +3.4 BLEU.

1 Introduction

Recent years have witnessed burgeoning development of statistical machine translation research, notably phrase-based (Koehn et al., 2003) and syntax-based approaches (Chiang, 2005; Galley et al., 2006; Liu et al., 2006). These approaches model sentence translation as a sequence of simple translation decisions, such as the application of a phrase translation in phrase-based methods or a grammar rule in syntax-based approaches. In order to simplify modelling, most MT models make an independence assumption, stating that the translation decisions in a derivation are independent of one another. This conflicts with the intuition behind phrase-based MT, namely that translation decisions should be dependent on context. On one hand, the use of phrases can memorize local context and hence helps to generate better translation compared to word-based models (Brown et al., 1993; Och and Ney, 2003). On the other hand, this mechanism requires each phrase to be matched strictly and to be used as a whole, which precludes the use of discontinuous phrases and leads to poor generalisation to unseen data (where large phrases tend not to match).

In this paper we propose a new model to drop the independence assumption, by instead modelling correlations between translation decisions, which we use to induce translation derivations from aligned sentences (akin to word alignment). We develop a Markov model over translation decisions, in which each decision is conditioned on previous n most recent decisions. Our approach employs a sophisticated Bayesian non-parametric prior, namely the hierarchical Pitman-Yor Process (Teh, 2006; Teh et al., 2006) to represent backoff from larger to smaller contexts. As a result, we need only use very simple translation units – primarily single words, but can still describe complex multi-word units through correlations between their component translation decisions. We further decompose the process of generating each target word into component factors: finishing the translating, jumping elsewhere in the source, emitting a target word and deciding the fertility of the source words.

Overall our model has the following features:

1. enabling model parameters to be shared between similar translation decisions, thereby obtaining more reliable statistics and generalizing better from small training sets.
2. learning a much richer set of translation fragments, such as *gapping* phrases, e.g., the translation for the German *werde . . . ankommen* in English is *will arrive . . .* .
3. providing a unifying framework spanning word-based and phrase-based model of translation, while incorporating explicit transla-

Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics, pages 333–342,
Sofia, Bulgaria, August 4-9 2013. ©2013 Association for Computational Linguistics

tion, insertion, deletion and reordering components.

We demonstrate our model on Chinese-English and Arabic-English translation datasets. The model produces uniformly better translations than those of a competitive phrase-based baseline, amounting to an improvement of up to 3.4 BLEU points absolute.

2 Related Work

Word based models have a long history in machine translation, starting with the venerable IBM translation models (Brown et al., 1993) and the hidden Markov model (Vogel et al., 1996). These models are still in wide-spread use today, albeit only as a preprocessing step for inferring word level alignments from sentence-aligned parallel corpora. They combine a number of factors, including distortion and fertility, which have been shown to improve word-alignment and translation performance over simpler models. Our approach is similar to these works, as we also develop a word-based model, and explicitly consider similar translation decisions, alignment jumps and fertility. We extend these works in two important respects: 1) while they assume a simple parameterisation by making *iid* assumptions about each translation factor, we instead allow for rich correlations by modelling sequences of translation decisions; and 2) we develop our model in the Bayesian framework, using a hierarchical Pitman-Yor Process prior with rich backoff semantics between high and lower order sequences of translation decisions. Together this results in a model with rich expressiveness but can still generalize well to unseen data.

More recently, a number of authors have proposed Markov models for machine translation. Vaswani et al. (2011) propose a rule Markov model for a tree-to-string model which models correlations between pairs of mininal rules, and use Kneser-Ney smoothing to alleviate the problems of data sparsity. Similarly, Crego et al. (2011) develop a bilingual language model which incorporates words in the source and target languages to predict the next unit, which they use as a feature in a translation system. This line of work was extended by Le et al. (2012) who develop a novel estimation algorithm based around discriminative projection into continuous spaces. Also relevant is Durrani et al. (2011), who present a sequence model of translation including reordering. Our work also uses bilingual information, using the source words as part of the conditioning context. In contrast to these approaches which primarily address the decoding problem, we focus on the learning problem of inferring alignments from parallel sentences. Additionally, we develop a full generative model using a Bayesian prior, and incorporate additional factors besides lexical items, namely jumps in the source and word fertility.

Another aspect of this paper is the implicit support for phrase-pairs that are discontinous in the source language. This idea has been developed explicitly in a number of previous approaches, in grammar based (Chiang, 2005) and phrase-based systems (Galley and Manning, 2010). The latter is most similar to this paper, and shows that discontinuous phrases compliment standard contiguous phrases, improving expressiveness and translation performance. Unlike their work, here we develop a complimentary approach by constructing a generative model which can induce these rich rules directly from sentence-aligned corpora.

3 Model

Given a source sentence, our model infers a latent derivation which produces a target translation and meanwhile gives a word alignment between the source and the target. We consider a process in which the target string is generated using a left-to-right order, similar to the decoding strategy used by phrase-based machine translation systems (Koehn et al., 2003). During this process we maintain a position in the source sentence, which can jump around to allow for different sentence ordering in the target vs. source languages. In contrast to phrase-based models, we use words as our basic translation unit, rather than multi-word phrases. Furthermore, we decompose the decisions involved in generating each target word to a number of separate factors, where each factor is modelled separately and conditioned on a rich history of recent translation decisions.

3.1 Markov Translation

Our model generates target translation left-to-right word by word. The generative process employs the following recursive procedure to construct the target sentence conditioned on the source:

```
$i \leftarrow 1$
while Not finished do
    Decide whether to finish the translation, $\xi_i$
```

Step	Source sentence				Translation					*finish*	*jump*	*emission*
0	__	Je	le	prends								
1		Je	le	prends	I					*no*	*monotone*	Je → I
2		Je	le	prends	I	’ll				*no*	*insert*	null → ’ll
3		Je	le	prends	I	’ll	take			*no*	*forward*	prends → take
4		Je	le	prends	I	’ll	take	that		*no*	*backward*	le → that
5		Je	le	prends	I	’ll	take	that	one	*no*	*stay*	le → one
6		Je	le	prends	I	’ll	take	that	one	*yes*		

Figure 1: Translation agenda of Je le prends → I ’ll take that one.

```
    if $\xi_i = \texttt{false}$ then
        Select a source word to jump to
        Emit a target word for the source word
    end if
    $i \leftarrow i + 1$
end while
```

In the generation of each target word, our model includes three separate factors: the binary *finish* decision, a *jump* decision to move to a different source word, and *emission* which translates or otherwise inserts a word in the target string. This generative process resembles the sequence of translation decisions considered by a standard MT decoder (Koehn et al., 2003), but note that our approach differs in that there is no constraint that all words are translated exactly once. Instead source words can be skipped or repeatedly translated. This makes the approach more suitable for learning alignments, e.g., to account for word fertilities (see §3.3), while also permitting inference using Gibbs sampling (§4).

More formally, we can express our probabilistic model as

$$
\begin{aligned}
p_{bs}(e_1^I, a_1^I | f_1^J) = & \prod_{i=1}^{I+1} p(\xi_i | f_{a i-n}^{i-1}, e_{i-n}^{i-1}) \\
& \times \prod_{i=1}^{I} p(\tau_i | f_{a i-n}^{i-1}, e_{i-n}^{i-1}) \\
& \times \prod_{i=1}^{I} p(e_i | \tau_i, f_{a i-n}^{i}, e_{i-n}^{i-1}) \qquad (1)
\end{aligned}
$$

where ξ_i is the finish decision for target position i, τ_i is the jump decision to source word f_{a_i} and $f_{a i-n}^{i}$ is the source words for target positions $i-n, i-n+1, ..., i$. Each of the three distributions (finish, jump and emission) is drawn respective from hierarchical Pitman-Yor Process priors, as described in Section 3.2.

The jump decision τ_i in Equation 1 demands further explanation. Instead of modelling jump distances explicitly, which poses problems for generalizing between different lengths of sentences and general parameter explosion, we consider a small handful of types of jump based on the distance between the current source word a_i and the previous source word a_{i-1}, i.e., $d_i = a_i - a_{i-1}$.[1] We bin jumps into five types:

a) *insert*;
b) *backward*, if $d_i < 0$;
c) *stay*, if $d_i = 0$;
d) *monotone*, if $d_i = 1$;
e) *forward*, if $d_i > 1$.

The special jump type *insert* handles null alignments, denoted $a_i = 0$ which licence spurious insertions in the target string.

To illustrate this translation process, Figure 1 shows the example translation <*Je le prends, I ’ll take that one*>. Initially we set the source position before the first source word *Je*. Then in step 1, we decide not to finish (finish=*no*), jump to source word *Je* and translate it as *I*. Next, we again decide not to finish, jump to the *null* source word and insert *’ll*. The process continues until in step 6 we elect to finish (finish=*yes*), at which point the translation is complete, with target string *I ’ll take that one*.

3.2 Hierarchical Pitman-Yor Process

The Markov assumption limits the context of each distribution to the n most recent translation decisions, which limits the number of model parameters. However for any non-trivial value $n > 0$, overfitting is a serious concern. We counter the problem of a large parameter space using a Bayesian non-parametric prior, namely the hierarchical Pitman-Yor Process (PYP). The PYP describes distributions over possibly infinite event spaces that follow a power law, with few events taking the majority of the probability mass and a long tail of less frequent events. We consider a hierarchical PYP, where a sequence of chained PYP

[1] For a target position aligned to null, we denote its source word as *null* and set its aligned source position as that of the previous target word that is aligned to non-null.

priors allow backoff from larger to smaller contexts such that our model can learn rich contextual models for known (large) contexts while also still being able to generalize well to unseen contexts (using smaller histories).

3.2.1 Pitman-Yor Process

A PYP (Pitman and Yor, 1997) is defined by its discount parameter $0 \leq a < 1$, strength parameter $b > -a$ and base distribution G_0. For a distribution drawn from a PYP, $G \sim \mathcal{PYP}(a, b, G_0)$, marginalising out G leads to a simple distribution which can be described using a variant of the Chinese Restaurant Process (CRP). In this analogy we imagine a restaurant has an infinite number of tables and each table can accommodate an infinite number of customers. Each customer (a sample from G) walks in one at a time and seats themselves at a table. Finally each table is served a communal dish (a draw from G_0), which is served to each customer seated at the table. The assignment of customers to tables is such that popular tables are more likely to be chosen, and this rich-get-richer dynamic produces power-law distributions with few events (the dishes at popular tables) dominating the distribution.

More formally, at time n a customer enters and selects a table k which is either a table having been seated ($1 \leq k \leq K^-$) or an empty table ($k = K^- + 1$) by

$$p(t_n = k|\boldsymbol{t}_{-n}) = \begin{cases} \frac{c^-_{\boldsymbol{t}k} - a}{n-1+b} & 1 \leq k \leq K^- \\ \frac{aK^- + b}{n-1+b} & k = K^- + 1 \end{cases}$$

where t_n is the table selected by the customer n, $\boldsymbol{t}_{-n}$ is the seating arrangement of previous $n-1$ customers, $c^-_{\boldsymbol{t}k}$ is the number of customers seated at table k in $\boldsymbol{t}_{-n}$ and $K^- = K(\boldsymbol{t}_{-n})$ is the number of tables in $\boldsymbol{t}_{-n}$.

If the customer sits at an empty table, a dish h is served to his table by the probability of $G_0(h)$, otherwise, he can only share with others the dish having been served to his table.[2] Overall, the probability of the customer being served a dish h is

$$\begin{aligned} p(o_n = h|\boldsymbol{t}_{-n}, \boldsymbol{o}_{-n}) &= \frac{c^-_{\boldsymbol{o}h} - aK^-_h}{n-1+b} \\ &+ \frac{(aK^- + b)}{n-1+b} G_0(h) \end{aligned}$$

where o_n is the dish served to the customer n, $\boldsymbol{o}_{-n}$ is the dish accommodation of previous $n-1$ customers, $c^-_{\boldsymbol{o}h}$ is the number of customers who are served with the dish h in $\boldsymbol{o}_{-n}$ and K^-_h is the number of tables served with the dish h in $\boldsymbol{t}_{-n}$.

The hierarchical PYP (hPYP; Teh (2006)) is an extension of the PYP in which the base distribution G_0 is itself a PYP distribution. This parent (base) distribution can itself have a PYP as a base distribution, giving rise to hierarchies of arbitrary depth. Like the PYP, inference under the hPYP can be also described in terms of CRP whereby each table in one restaurant corresponds to a dish in the next deeper level, and is said to share the same dish. Whenever an empty table is seated in one level, a customer must enter the restaurant in the next deeper level and find a table to sit. This process continues until the customer is assigned a shared table or the deepest level of the hierarchy is reached. A similar process occurs when a customer leaves, where newly emptied tables must be propagated up the hierarchy in the form of departing customers. There is not space for a complete treatment of the hPYP and the particulars of inference; we refer the interested reader to Teh (2006).

3.2.2 A Hierarchical PYP Translation Model

We draw the distributions for the various translation factors from respective hierarchical PYP priors, as shown in Figure 2 for the finish, jump and emission factors. For the emission factor (Figure 2c), we draw the target word e_i from a distribution conditioned on the last two source and target words, as well as the current source word, f_{a_i} and the current jump type τ_i. Here the draw of a target word corresponds to a customer entering and which target word to emit corresponds to which dish to be served to the customer in the CRP. The hierarchical prior encodes a backoff path in which the jump type is dropped first, followed by pairs of source and target words from least recent to most recent. The final backoff stages drop the current source word, terminating with the uniform base distribution over the target vocabulary V.

The distributions over the other two factors in Figure 2 follow a similar pattern. Note however that these distributions don't condition on the current source word, and consequently have fewer levels of backoff. The terminating base distribution for the finish factor is a uniform distribution with equal probability for finishing versus continuing. The jump factor has an additional conditioning variable t which encodes whether the previous alignment is near the start or end of the source sentence. This information affects which of the jump values are legal from the current position, such

[2] We also say the customer is served with this dish.

$$\begin{aligned}
\xi_i | f^{i-1}_{a_{i-2}}, e^{i-1}_{i-2} &\sim G^{\xi}_{f^{i-1}_{a_{i-2}}, e^{i-1}_{i-2}} \\
G^{\xi}_{f^{i-1}_{a_{i-2}}, e^{i-1}_{i-2}} &\sim \mathcal{PYP}(a^{\xi}_3, b^{\xi}_3, G^{\xi}_{f_{a_{i-1}}, e_{i-1}}) \\
G^{\xi}_{f_{a_{i-1}}, e_{i-1}} &\sim \mathcal{PYP}(a^{\xi}_2, b^{\xi}_2, G^{\xi}) \\
G^{\xi} &\sim \mathcal{PYP}(a^{\xi}_1, b^{\xi}_1, G^{\xi}_0) \\
G^{\xi}_0 &\sim \mathcal{U}(\frac{1}{2})
\end{aligned}$$

(a) Finish factor

$$\begin{aligned}
\tau_i | f^{i-1}_{a_{i-2}}, e^{i-1}_{i-2}, t &\sim G^{\tau}_{f^{i-1}_{a_{i-2}}, e^{i-1}_{i-2}, t} \\
G^{\tau}_{f^{i-1}_{a_{i-2}}, e^{i-1}_{i-2}, t} &\sim \mathcal{PYP}(a^{\tau}_3, b^{\tau}_3, G^{\tau}_{f_{a_{i-1}}, e_{i-1}, t}) \\
G^{\tau}_{f_{a_{i-1}}, e_{i-1}, t} &\sim \mathcal{PYP}(a^{\tau}_2, b^{\tau}_2, G^{\tau}_t) \\
G^{\tau}_t &\sim \mathcal{PYP}(a^{\tau}_1, b^{\tau}_1, G^{\tau}_{0,t}) \\
G^{\tau}_{0,t} &\sim \mathcal{U}
\end{aligned}$$

(b) Jump factor

$$\begin{aligned}
e_i | \tau_i, f^{i}_{a_{i-2}}, e^{i-1}_{i-2} &\sim G^{e}_{\tau_i, f^{i}_{a_{i-2}}, e^{i-1}_{i-2}} \\
G^{e}_{\tau_i, f^{i}_{a_{i-2}}, e^{i-1}_{i-2}} &\sim \mathcal{PYP}(a^{e}_5, b^{e}_5, G^{e}_{f^{i}_{a_{i-2}}, e^{i-1}_{i-2}}) \\
G^{e}_{f^{i}_{a_{i-2}}, e^{i-1}_{i-2}} &\sim \mathcal{PYP}(a^{e}_4, b^{e}_4, G^{e}_{f^{i}_{a_{i-1}}, e_{i-1}}) \\
G^{e}_{f^{i}_{a_{i-1}}, e_{i-1}} &\sim \mathcal{PYP}(a^{e}_3, b^{e}_3, G^{e}_{f_{a_i}}) \\
G^{e}_{f_{a_i}} &\sim \mathcal{PYP}(a^{e}_2, b^{e}_2, G^{e}) \\
G^{e} &\sim \mathcal{PYP}(a^{e}_1, b^{e}_1, G^{e}_0) \\
G^{e}_0 &\sim \mathcal{U}(\frac{1}{|V|})
\end{aligned}$$

(c) Emission factor

Figure 2: Distributions over the translation factors and their hierarchical priors.

that a jump could not go outside the bounds of the source sentence. Accordingly we maintain separate distributions for each setting, and each has a different uniform base distribution parameterized according to the number of possible jump types.

3.3 Fertility

For each target position, our Markov model may select a source word which has been covered, which means a source word may be linked to several target positions. Therefore, we introduce *fertility* to denote the number of target positions a source word is linked to in a sentence pair. Brown et al. (1993) have demonstrated the usefulness of fertility in probability estimation: IBM models 3–5 exhibit large improvements over models 1–2. On these grounds, we include fertility to produce our *advanced* model,

$$p_{ad}(e_1^I, a_1^I | f_1^J) = p_{bs}(e_1^I, a_1^I | f_1^J) \prod_{j=1}^{J} p(\phi_j | f_{j-n}^j) \quad (2)$$

where ϕ_j is the fertility of source word f_j in the sentence pair $< f_1^J, e_1^I >$ and p_{bs} is the basic model defined in Eq. 1. In order to avoid problems of data sparsity, we bin fertility into three types, a) *zero*, if $\phi = 0$; b) *single*, if $\phi = 1$; and c) *multiple*, if $\phi > 1$.

We draw the fertility variables from a hierarchical PYP distribution, using three levels of backoff,

$$\begin{aligned}
\phi_j | f_{j-1}^j &\sim G^{\phi}_{f_{j-1}^j} \\
G^{\phi}_{f_{j-1}^j} &\sim \mathcal{PYP}(a_3^{\phi}, b_3^{\phi}, G^{\phi}_{f_j}) \\
G^{\phi}_{f_j} &\sim \mathcal{PYP}(a_2^{\phi}, b_2^{\phi}, G^{\phi}) \\
G^{\phi} &\sim \mathcal{PYP}(a_1^{\phi}, b_1^{\phi}, G_0^{\phi}) \\
G_0^{\phi} &\sim \mathcal{U}(\frac{1}{3})
\end{aligned}$$

where we condition the fertility of each word token on the token to its left, which we drop during the first stage of backoff to simple word-based fertility. The last level of backoff further generalises to a shared fertility across all words. In this way we gain the benefits of local context on fertility, while including more general levels to allow wider applicability.

4 Gibbs Sampling

To train the model, we use Gibbs sampling, a Markov Chain Monte Carlo (MCMC) technique for posterior inference. Specifically we seek to infer the latent sequence of translation decisions given a corpus of sentence pairs. Given the structure of our model, a word alignment uniquely specifies the translation decisions and the sequence follows the order of the target sentence left to right. Our Gibbs sampler operates by sampling an update to the alignment of each target word in the corpus. It visits each sentence pair in the corpus in a random order and resamples the alignments for each target position as follows. First we discard the alignment to the current target word and decrement the counts of all factors affected by this alignment in their top level distributions (which will percolate down to the lower restaurants). Next we calculate posterior probabilities for all possible alignment to this target word based on the table occupancies in the hPYP. Finally we draw an alignment and increment the table counts for the translation decisions affected by the new alignment.

More specifically, we consider sampling from Equation 2 with $n = 2$. When changing the alignment to a target word e_i from j' to j, *t*he finish, *jump* and *emission* for three target positions $i, i+1, i+2$ and *fertility* for two source positions j, j' may be affected. This leads to the following

decrement	increment
ξ(no $\mid$ null, ’ll, Je, I)	ξ(no $\mid$ null, ’ll, Je, I)
ξ(no $\mid$ p..s, take, null, ’ll)	ξ(no $\mid$ Je, take, null, ’ll)
ξ(no $\mid$ le, that, p..s, take)	ξ(no $\mid$ le, that, Je, take)
$\tau(f$ $\mid$ null, ’ll, Je, I)	$\tau(s\mid$ null, ’ll, Je, I)
$\tau(b$ $\mid$ p..s, take, null, ’ll)	$\tau(m\mid$ Je, take, null, ’ll)
$\tau(s$ $\mid$ le, that, p..s, take)	$\tau(s\mid$ le, that, Je, take)
e(take $\mid f$, p..s, null, ’ll, Je, I)	e(take $\mid s$, Je, null, ’ll, Je, I)
e(that $\mid b$, le, p..s, take, null, ’ll)	e(that $\mid m$, le, Je, take, null, ’ll)
e(one $\mid s$, le, le, that, p..s, take)	e(one $\mid s$, le, le, that, Je, take)
ϕ(single $\mid$ p..s, le)	ϕ(multiple $\mid$ Je, <s>)

Table 1: The count update when changing the aligned source word of take *from* prends *to* Je *in Figure 1. Key:* f–forward s–stay b–backward m–monotone p..s–prends.

posterior probability

$$p(a_i = j|\boldsymbol{t}_{-i}, \boldsymbol{o}_{-i}) \propto \prod_{l=i}^{i+2} p(\xi_l)p(\tau_l)p(e_l) \\ \times \frac{p(\phi_j + 1)p(\phi_{j'} - 1)}{p(\phi_j)p(\phi_{j'})} \quad (3)$$

where $\phi_j, \phi_{j'}$ are the fertilities before changing the link and for brevity we omit the conditioning contexts. For example, in Figure 1, we sample for target word *take* and change the aligned source word from *prends* to *Je*, then the items for which we need to decrement and increment the counts by one are shown in Table 1 and the posterior probability corresponding to the new alignment is the product of the hierarchical PYP probabilities of all increment items divided by the probability of the fertility of *prends* being *single*.

Maintaining the current state of the hPYP as events are incremented and decremented is non-trivial and the naive approach requires significant book-keeping and has poor runtime behaviour. For this we adopt the approach of Blunsom et al. (2009b), who present a method for maintaining table counts without needing to record the table assignments for each translation decision. Briefly, this algorithm samples the table assignment during the increment and decrement operations, which is then used to maintain aggregate table statistics. This can be done efficiently and without the need for explicit table assignment tracking.

4.1 Hyperparameter Inference

In our model, we treat all hyper-parameters $\{(a^x, b^x), x \in (\xi, \tau, e, \phi)\}$ as latent random variables rather than fixed parameters. This means our model is parameter free, and requires no user intervention when adapting to different data sets. For the discount parameter, we employ a uniform Beta distribution $a^x \sim \mathrm{Beta}(1, 1)$ while for the strength parameter, we employ a vague Gamma distribution $b^x \sim \mathrm{Gamma}(10, 0.1)$. All restaurants in the same level share the same hyper-prior and the hyper-parameters for all levels are resampled using slice sampling (Johnson and Goldwater, 2009) every 10 iterations.

4.2 Parallel Implementation

As mentioned above, the hierarchical PYP takes into consideration a rich history to evaluate the probabilities of translation decisions. But this leads to difficulties when applying the model to large data sets, particularly in terms of tracking the table and customer counts. We apply the technique from Blunsom et al. (2009a) of using multiple processors to perform approximate Gibbs sampling which they showed achieved equivalent performance to the exact Gibbs sampler. Each process performs sampling on a subset of the corpus using local counts, and communicates changes to these counts after each full iteration. All the count deltas are then aggregated by each process to refresh the counts at the end of each iteration. In this way each process uses slightly “out-of-date” counts, but can process the data independently of the other processes. We found that this approximation improved the runtime significantly with no noticeable effect on accuracy.

5 Experiments

In principle our model could be directly used as a MT decoder or as a feature in a decoder. However in this paper we limit our focus to inducing word alignments, i.e., by using the model to infer alignments which are then used in a standard phrase-based translation pipeline. We leave full decoding for later work, which we anticipate would further improve performance by exploiting gapping phrases and other phenomena that implicitly form part of our model but are not represented in the phrase-based decoder. Decoding under our model would be straight-forward in principle, as the generative process was designed to closely parallel the search procedure in the phrase-based model.[3]

Three data sets were used in the experiments: two Chinese to English data sets on small (IWSLT) and larger corpora (FBIS), and Arabic

[3] However the reverse translation probability would be intractable, as this does not decompose following a left-to-right generation order in the target language.

to English translation. Our experiments seek to test how the model compares to a GIZA++ baseline, quantifies the effect of each factor in the probabilistic model (i.e., jump, fertility), and the effect of different initialisations of the sampler. We present results on translation quality and word alignment.

5.1 Data Setup

The Markov order of our model in all experiments was set to $n = 2$, as shown in Equation 2. For each data set, Gibbs sampling was performed on the training set in each direction (source-to-target and target-to-source), initialized using GIZA++.[4] We used the *grow* heuristic to combine the GIZA++ alignments in both directions (Koehn et al., 2003), which we then intersect with the predictions of GIZA++ in the relevant translation direction. This initialisation setup gave the best results (we compare other initialisations in §5.2). The two Gibbs samplers were "burned in" for the first 1000 iterations, after which we ran a further 500 iterations selecting every 50th sample. A phrase table was constructed using these 10 sets of multiple alignments after combining each pair of directional alignments using the *grow-diag-final* heuristic. Using multiple samples in this way constitutes Monte Carlo averaging, which provides a better estimate of uncertainty cf. using a single sample.[5]

The alignment used for the baseline results was produced by combining bidirectional GIZA++ alignments using the *grow-diag-final* heuristic. We used the Moses machine translation decoder (Koehn et al., 2007), using the default features and decoding settings. We compared the performance of Moses using the alignment produced by our model and the baseline alignment, evaluating translation quality using BLEU (Papineni et al., 2002) with case-insensitive n-gram matching with $n = 4$. We used minimum error rate training (Och, 2003) to tune the feature weights to maximise the BLEU score on the development set.

5.2 IWSLT Corpus

The first experiments are on the IWSLT data set for Chinese-English translation. The training data consists of 44k sentences from the tourism and travel domain. For the development set we use both ASR devset 1 and 2 from IWSLT 2005, and for the test set we use the IWSLT 2005 test set. The language model is a 3-gram language model trained using the SRILM toolkit (Stolcke, 2002) on the English side of the training data. Because the data set is small, we performed Gibbs sampling on a single processor.

System	Dev	IWSLT05
baseline	45.78	49.98
Markov+f_s+e	49.13	51.54
Markov+f_s+e+j	49.68	52.55
Markov+f_s+e+j+f_t	51.32	53.41

Table 2: Impact of adding factors to our Markov model, showing BLEU scores on IWSLT. Key: f_s*–finish* e*–emission* j*–jump* f_t*–fertility.*

First we check the effect of the model factors *jump* and *fertility*. Both *emission* and *finish* factors are indispensable to the generative translation process, and consequently these two factors are included in all runs. Table 2 shows translation result for various models, including a baseline and our Markov model with different combinations of factors. Note that even the simplest Markov model far outperforms the GIZA++ baseline (+1.5 BLEU) despite the baseline (IBM model 4) including a number of advanced features (e.g., *jump*, *fertility*) that are not present in the basic Markov model. This improvement is a result of the Markov model making use of rich bilingual contextual information coupled with sophisticated backoff, as opposed to GIZA++ which considers much more local events, with nothing larger than word-class bigrams. Our model shows large improvements as the extra factors are included. *Jump* yields an improvement of +1 BLEU by capturing consistent reordering patterns. Adding *fertility* results in a further +1 BLEU point improvement. Like the IBM models, our approach allows each source word to produce any number of target words. This capacity allows for many non-sensical alignments such as dropping many source words, or aligning single source words to several target words. Explicitly modelling fertility allows for more consistent alignments, especially for special words such as punctuation which usually have a fertility of one.

Next we check the stability of our model with different initialisations. We compare different combination techniques for merging the GIZA++ alignments: *grow-diag-final* (denoted as *gdf*), *intersection* and *grow*. Table 3 shows that the different initialisations have only a small effect on

[4] All GIZA++ alignments used in our experiments were produced by IBM model4.

[5] The effect on translation scores is modest, roughly amounting to +0.2 BLEU versus using a single sample.

system	gdf	intersection	grow
baseline	49.98	48.44	50.11
our model	52.96	52.79	53.41

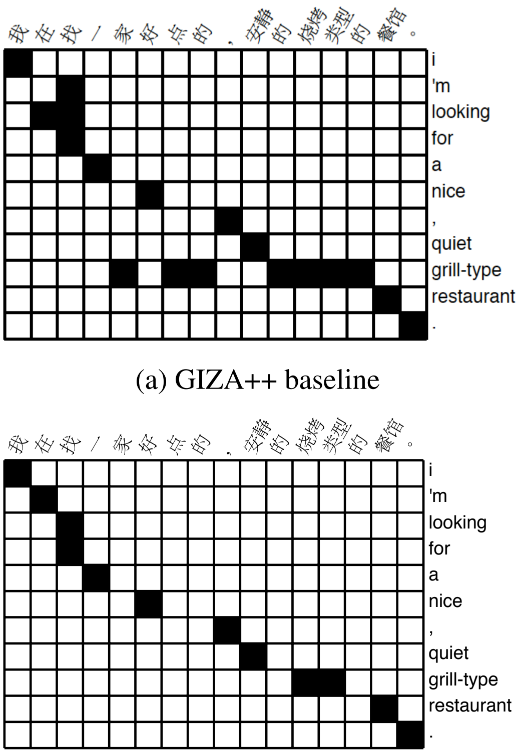
Table 3: Machine translation performance in BLEU % on the IWSLT 2005 Chinese-English test set. The Gibbs samplers were initialized with three different alignments, shown as columns.

the results of our model. While the baseline results vary by up to 1.7 BLEU points for the different alignments, our Markov model provided more stable results with the biggest difference of 0.6. Among the three initialisations, we get the best result with the initialisation of *grow*. *Gdf* often introduces alignment links involving function words which should instead be aligned to null. *Intersection* includes many fewer alignments, typically only between content words, and the sparsity means that words can only have a fertility of either 0 or 1. This leads to the initialisation being a strong mode which is difficult to escape from during sampling. Despite this problem, it has only a mild negative effect on the performance of our model, which is probably due to improvements in the alignments for words that truly should be dropped or aligned only to one word. *Grow* provides a good compromise between *gdf* and *intersection*, and we use this initialisation in all our subsequent experiments.

Figure 3 shows an example comparing alignments produced by our model and the GIZA++ baseline, in both cases after combining the two directional models. Note that GIZA++ has linked many function words which should be left unaligned, by using rare English terms as garbage collectors. Consequently this only allows for the extraction of few large phrase-pairs (e.g. <在 找, 'm looking for>) and prevents the extraction of some good phrases (e.g. <烧烤 类型 的, grill-type>, for "家" and "点 的" are wrongly aligned to "grill-type"). In contrast, our model better aligns the function words, such that many more useful phrase pairs can be extracted, i.e., <在, 'm>, <找, looking for>, <烧烤 类型, grill-type> and their combinations with neighbouring phrase pairs.

5.3 FBIS Corpus

Theoretically, Bayesian models should outperform maximum likelihood approaches on small data sets, due to their improved modelling of uncertainty. For larger datasets, however, the difference between the two techniques should narrow. Hence one might expect that upon moving to larger translation datasets our gains might evaporate. This chain of reasoning ignores the fact that our model is considerably richer than the baseline IBM models, in that we model rich contextual correlations between translation decisions, and consequently our approach has a lower inductive bias. For this reason our model should continue to improve with more data, by inferring better estimates of translation decision n-grams. A caveat though is that inference by sampling becomes less efficient on larger data sets due to stronger modes, requiring more iterations for convergence.



(a) GIZA++ baseline

(b) our model

Figure 3: Comparison of an alignment inferred by the baseline vs. our approach.

To test whether our improvements carry over to larger datasets, we assess the performance of our model on the FBIS Chinese-English data set. Here the training data consists of the non-UN portions and non-HK Hansards portions of the NIST training corpora distributed by the LDC, totalling 303k sentence pairs with 8m and 9.4m words of Chinese and English, respectively. For the development set we use the NIST 2002 test set, and evaluate performance on the test sets from NIST 2003

	NIST02	NIST03	NIST05
baseline	33.31	30.09	29.01
our model	33.83	31.02	30.23

Table 4: Translation performance on Chinese to English translation, showing BLEU% for models trained on the FBIS data set.

	F1%	NIST02	NIST03	NIST05
baseline	64.9	57.00	48.75	48.93
our model	65.7	57.14	49.49	48.96

Table 5: Translation performance on Arabic to English translation, showing BLEU%. Also shown is word-alignment alignment accuracy.

and 2005. The language model is a 3-gram LM trained on Xinhua portion of the Gigaword corpus using the SRILM toolkit with modified Kneser-Ney smoothing. As the FBIS data set is large, we employed 3-processor MPI for each Gibbs sampler, which ran in half the time compared to using a single processor.

Table 4 shows the results on the FBIS data set. Our model outperforms the baseline on both test sets by about 1 BLEU. This provides evidence that our model performs well in the large data setting, with our rich modelling of context still proving useful. The non-parametric nature of the model allows for rich dynamic backoff behaviour such that it can learn accurate models in both high and low data scenarios.

5.4 Arabic English translation

Translation between Chinese and English is very difficult, particularly due to word order differences which are not handled well by phrase-based approaches. In contrast Arabic to English translation needs less reordering, and phrase-based models produce better translations. This translation task is a good test for the generality of our approach. Our Ar-En training data comprises several LDC corpora,[6] using the same experimental setup as in Blunsom et al. (2009a). Overall there are 276k sentence pairs and 8.21m and 8.97m words in Arabic and English, respectively. We evaluate on the NIST test sets from 2003 and 2005, and the 2002 test set was used for MERT training.

Table 5 shows the results. On all test sets our approach outperforms the baseline, and for the NIST03 test set the improvement is substantial, with a +0.74 BLEU improvement. In general the improvements are more modest than for the Chinese-English results above. We suggest that this is due to the structure of Arabic-English translation better suiting the modelling assumptions behind IBM model 4, particularly its bias towards monotone translations. Consequently the additional context provided by our model is less important. Table 5 also reports alignment results on manually aligned Ar-En sentence pairs,[7] measuring the F1 score for the GIZA++ baseline alignments and the alignment from the final sample with our model.[8] Our model outperforms the baseline, although the improvement is modest.

6 Conclusions and Future Work

This paper proposes a word-based Markov model of translation which correlates translation decisions by conditioning on recent decisions, and incorporates a hierarchical Pitman-Yor process prior permitting elaborate backoff behaviour. The model can learn sequences of translation decisions, akin to phrases in standard phrase-based models, while simultaneously learning word level phenomena. This mechanism generalises the concept of phrases in phrase-based MT, while also capturing richer phenomena such as gapping phrases in the source. Experiments show that our model performs well both on the small and large datasets for two different translation tasks, consistently outperforming a competitive baseline. In this paper the model was only used to infer word alignments; in future work we intend to develop a decoding algorithm for directly translating with the model.

Acknowledgements

This work was supported by the EPSRC (grant EP/I034750/1).

References

Phil Blunsom, Trevor Cohn, Chris Dyer, and Miles Osborne. 2009a. A Gibbs sampler for phrasal synchronous grammar induction. In *Proc. of ACL-IJCNLP*, pages 782–790.

[6] LDC2004E72, LDC2004T17, LDC2004T18, LDC2006T02

[7] LDC2012T16

[8] Directional alignments are intersected using the grow-diag-final heuristic.

Phil Blunsom, Trevor Cohn, Sharon Goldwater, and Mark Johnson. 2009b. A note on the implementation of hierarchical dirichlet processes. In *Proc. of ACL-IJCNLP*, pages 337–340.

Peter E. Brown, Stephen A. Della Pietra, Vincent J. Della Pietra, and Robert L. Mercer. 1993. The mathematics of statistical machine translation: Parameter estimation. *Computational Linguistics*, 19:263–331.

David Chiang. 2005. A hierarchical phrase-based model for statistical machine translation. In *Proc. of ACL*, pages 263–270.

Josep Maria Crego, François Yvon, and José B. Mariño. 2011. Ncode: an open source bilingual n-gram SMT toolkit. *Prague Bull. Math. Linguistics*, 96:49–58.

Nadir Durrani, Helmut Schmid, and Alexander Fraser. 2011. A joint sequence translation model with integrated reordering. In *Proc. of ACL:HLT*, pages 1045–1054.

Michel Galley and Christopher D. Manning. 2010. Accurate non-hierarchical phrase-based translation. In *Proc. of NAACL*, pages 966–974.

Michel Galley, Jonathan Graehl, Kevin Knight, Daniel Marcu, Steve DeNeefe, Wei Wang, and Ignacio Thayer. 2006. Scalable inference and training of context-rich syntactic translation models. In *Proc. of ACL*, pages 961–968.

Mark Johnson and Sharon Goldwater. 2009. Improving nonparameteric bayesian inference: experiments on unsupervised word segmentation with adaptor grammars. In *Proc. of HLT-NAACL*, pages 317–325.

Philipp Koehn, Franz J. Och, and Daniel Marcu. 2003. Statistical phrase-based translation. In *Proc. of HLT-NAACL*, pages 127–133.

Philipp Koehn, Hieu Hoang, Alexandra Birch Mayne, Christopher 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 ACL*.

Hai-Son Le, Alexandre Allauzen, and François Yvon. 2012. Continuous space translation models with neural networks. In *Proc. of NAACL*, pages 39–48.

Yang Liu, Qun Liu, and Shouxun Lin. 2006. Tree-to-string alignment template for statistical machine translation. In *Proc. of COLING-ACL*, pages 609–616, July.

Frans J. Och and Hermann Ney. 2003. A systematic comparison of various statistical alignment models. *Computational Linguistics*, 29:19–51.

Frans J. Och. 2003. Minimum error rate training in statistical machine translation. In *Proc. of ACL*, pages 160–167.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. BLEU: A method for automatic evaluation of machine translation. In *Proc. of ACL*, pages 311–318.

Jim Pitman and Marc Yor. 1997. The two-parameter poisson-dirichlet distribution derived from a stable subordinator. *The Annals of Probability*, 25(2):855–900.

Andreas Stolcke. 2002. SRILM: An extensible language modeling toolkit. In *Proc. of ICSLP*.

Y. W. Teh, M. I. Jordan, M. J. Beal, and D. M. Blei. 2006. Hierarchical Dirichlet processes. *Journal of the American Statistical Association*, 101(476):1566–1581.

Yee Whye Teh. 2006. A hierarchical Bayesian language model based on Pitman-Yor processes. In *Proc. of ACL*, pages 985–992.

Ashish Vaswani, Haitao Mi, Liang Huang, and David Chiang. 2011. Rule markov models for fast tree-to-string translation. In *Proc. of ACL*, pages 856–864.

Stephan Vogel, Hermann Ney, and Christoph Tillmann. 1996. HMM-based word alignment in statistical translation. In *Proc. of COLING*, pages 836–841.