A Gibbs Sampler for Phrasal Synchronous Grammar Induction

Phil Blunsom[*]
pblunsom@inf.ed.ac.uk

Trevor Cohn[*]
tcohn@inf.ed.ac.uk

Chris Dyer[†]
redpony@umd.edu

Miles Osborne[*]
miles@inf.ed.ac.uk

[*]Department of Informatics
University of Edinburgh
Edinburgh, EH8 9AB, UK

[†]Department of Linguistics
University of Maryland
College Park, MD 20742, USA

Abstract

We present a phrasal synchronous grammar model of translational equivalence. Unlike previous approaches, we do not resort to heuristics or constraints from a word-alignment model, but instead directly induce a synchronous grammar from parallel sentence-aligned corpora. We use a hierarchical Bayesian prior to bias towards compact grammars with small translation units. Inference is performed using a novel Gibbs sampler over synchronous derivations. This sampler side-steps the intractability issues of previous models which required inference over derivation forests. Instead each sampling iteration is highly efficient, allowing the model to be applied to larger translation corpora than previous approaches.

1 Introduction

The field of machine translation has seen many advances in recent years, most notably the shift from word-based (Brown et al., 1993) to phrase-based models which use token n-grams as translation units (Koehn et al., 2003). Although very few researchers use word-based models for translation per se, such models are still widely used in the training of phrase-based models. These word-based models are used to find the latent word-alignments between bilingual sentence pairs, from which a weighted string transducer can be induced (either finite state (Koehn et al., 2003) or synchronous context free grammar (Chiang, 2007)). Although wide-spread, the disconnect between the translation model and the alignment model is artificial and clearly undesirable. Word-based models are incapable of learning translational equivalences between non-compositional phrasal units, while the algorithms used for inducing weighted transducers from word-alignments are based on heuristics with little theoretical justification. A model which can fulfil both roles would address both the practical and theoretical short-comings of the machine translation pipeline.

The machine translation literature is littered with various attempts to learn a phrase-based string transducer directly from aligned sentence pairs, doing away with the separate word alignment step (Marcu and Wong, 2002; Cherry and Lin, 2007; Zhang et al., 2008b; Blunsom et al., 2008). Unfortunately none of these approaches resulted in an unqualified success, due largely to intractable estimation. Large training sets with hundreds of thousands of sentence pairs are common in machine translation, leading to a parameter space of billions or even trillions of possible bilingual phrase-pairs. Moreover, the inference procedure for each sentence pair is non-trivial, proving NP-complete for learning phrase based models (DeNero and Klein, 2008) or a high order polynomial ($O(|\mathbf{f}|^3|\mathbf{e}|^3)$)[1] for a sub-class of weighted synchronous context free grammars (Wu, 1997). Consequently, for such models both the parameterisation and approximate inference techniques are fundamental to their success.

In this paper we present a novel SCFG translation model using a non-parametric Bayesian formulation. The model includes priors to impose a bias towards small grammars with few rules, each of which is as simple as possible (e.g., terminal productions consisting of short phrase pairs). This explicitly avoids the degenerate solutions of maximum likelihood estimation (DeNero et al., 2006), without resort to the heuristic estimator of Koehn et al. (2003). We develop a novel Gibbs sampler to perform inference over the latent synchronous derivation trees for our training instances. The sampler reasons over the infinite space of possible translation units without recourse to arbitrary restrictions (e.g., constraints drawn from a word-alignment (Cherry and Lin, 2007; Zhang et al., 2008b) or a grammar fixed a priori (Blunsom et al.,

[1] $\mathbf{f}$ and $\mathbf{e}$ are the input and output sentences respectively.

2008)). The sampler performs local edit operations to nodes in the synchronous trees, each of which is very fast, leading to a highly efficient inference technique. This allows us to train the model on large corpora without resort to punitive length limits, unlike previous approaches which were only applied to small data sets with short sentences.

This paper is structured as follows: In Section 3 we argue for the use of efficient sampling techniques over SCFGs as an effective solution to the modelling and scaling problems of previous approaches. We describe our Bayesian SCFG model in Section 4 and a Gibbs sampler to explore its posterior. We apply this sampler to build phrase-based and hierarchical translation models and evaluate their performance on small and large corpora.

2 Synchronous context free grammar

A synchronous context free grammar (SCFG, (Lewis II and Stearns, 1968)) generalizes context-free grammars to generate strings concurrently in two (or more) languages. A string pair is generated by applying a series of paired rewrite rules of the form, $X \rightarrow \langle \mathbf{e}, \mathbf{f}, \mathbf{a} \rangle$, where X is a non-terminal, $\mathbf{e}$ and $\mathbf{f}$ are strings of terminals and non-terminals and $\mathbf{a}$ specifies a one-to-one alignment between non-terminals in $\mathbf{e}$ and $\mathbf{f}$. In the context of SMT, by assigning the source and target languages to the respective sides of a probabilistic SCFG it is possible to describe translation as the process of parsing the source sentence, which induces a parallel tree structure and translation in the target language (Chiang, 2007). Figure 1 shows an example derivation for Japanese to English translation using an SCFG. For efficiency reasons we only consider binary or ternary branching rules and don't allow rules to mix terminals and non-terminals. This allows our sampler to more efficiently explore the space of grammars (Section 4.2), however more expressive grammars would be a straightforward extension of our model.

3 Related work

Most machine translation systems adopt the approach of Koehn et al. (2003) for 'training' a phrase-based translation model.[2] This method starts with a word-alignment, usually the latent state of an unsupervised word-based aligner such as GIZA++. Various heuristics are used to combine source-to-target and target-to-source alignments, after which a further heuristic is used to read off phrase pairs which are 'consistent' with the alignment. Although efficient, the sheer number of somewhat arbitrary heuristics makes this approach overly complicated.

Grammar fragment:

$$
\begin{aligned}
X &\rightarrow \langle \mathrm{X}_{\boxed{1}}\ \mathrm{X}_{\boxed{2}}\ \mathrm{X}_{\boxed{3}},\ \mathrm{X}_{\boxed{1}}\ \mathrm{X}_{\boxed{3}}\ \mathrm{X}_{\boxed{2}} \rangle \\
X &\rightarrow \langle \textit{John-ga},\ \textit{John} \rangle \\
X &\rightarrow \langle \textit{ringo-o},\ \textit{an apple} \rangle \\
X &\rightarrow \langle \textit{tabeta},\ \textit{ate} \rangle
\end{aligned}
$$

Sample derivation:

$$
\begin{aligned}
& \langle \mathrm{S}_{\boxed{1}}, \mathrm{S}_{\boxed{1}} \rangle \Rightarrow \langle \mathrm{X}_{\boxed{2}},\ \mathrm{X}_{\boxed{2}} \rangle \\
\Rightarrow\ & \langle \mathrm{X}_{\boxed{3}}\ \mathrm{X}_{\boxed{4}}\ \mathrm{X}_{\boxed{5}},\ \mathrm{X}_{\boxed{3}}\ \mathrm{X}_{\boxed{5}}\ \mathrm{X}_{\boxed{4}} \rangle \\
\Rightarrow\ & \langle \textit{John-ga}\ \mathrm{X}_{\boxed{4}}\ \mathrm{X}_{\boxed{5}},\ \textit{John}\ \mathrm{X}_{\boxed{5}}\ \mathrm{X}_{\boxed{4}} \rangle \\
\Rightarrow\ & \langle \textit{John-ga ringo-o}\ \mathrm{X}_{\boxed{5}},\ \textit{John}\ \mathrm{X}_{\boxed{5}}\ \textit{an apple} \rangle \\
\Rightarrow\ & \langle \textit{John-ga ringo-o tabeta},\ \textit{John ate an apple} \rangle
\end{aligned}
$$

Figure 1: A fragment of an SCFG with a ternary non-terminal expansion and three terminal rules.

A number of authors have proposed alternative techniques for directly inducing phrase-based translation models from sentence aligned data. Marcu and Wong (2002) proposed a phrase-based alignment model which suffered from a massive parameter space and intractable inference using expectation maximisation. Taking a different tack, DeNero et al. (2008) presented an interesting new model with inference courtesy of a Gibbs sampler, which was better able to explore the full space of phrase translations. However, the efficacy of this model is unclear due to the small-scale experiments and the short sampling runs. In this work we also propose a Gibbs sampler but apply it to the polynomial space of derivation trees, rather than the exponential space of the DeNero et al. (2008) model. The restrictions imposed by our tree structure make sampling considerably more efficient for long sentences.

Following the broad shift in the field from finite state transducers to grammar transducers (Chiang, 2007), recent approaches to phrase-based alignment have used synchronous grammar formalisms permitting polynomial time inference (Wu, 1997;

[2] We include grammar based transducers, such as Chiang (2007) and Marcu et al. (2006), in our definition of phrase-based models.

Cherry and Lin, 2007; Zhang et al., 2008b; Blunsom et al., 2008). However this asymptotic time complexity is of high enough order ($O(|\mathbf{f}|^3|\mathbf{e}|^3)$) that inference is impractical for real translation data. Proposed solutions to this problem include imposing sentence length limits, using small training corpora and constraining the search space using a word-alignment model or parse tree. None of these limitations are particularly desirable as they bias inference. As a result phrase-based alignment models are not yet practical for the wider machine translation community.

4 Model

Our aim is to induce a grammar from a training set of sentence pairs. We use Bayes' rule to reason under the posterior over grammars, $P(\mathbf{g}|\mathbf{x}) \propto P(\mathbf{x}|\mathbf{g})P(\mathbf{g})$, where $\mathbf{g}$ is a weighted SCFG grammar and $\mathbf{x}$ is our training corpus. The likelihood term, $P(\mathbf{x}|\mathbf{g})$, is the probability of the training sentence pairs under the grammar, while the prior term, $P(\mathbf{g})$, describes our initial expectations about what consitutes a plausible grammar. Specifically we incorporate priors encoding our preference for a briefer and more succinct grammar, namely that: (a) the grammar should be small, with few rules rewriting each non-terminal; and (b) terminal rules which specify phrasal translation correspondence should be small, with few symbols on their right hand side.

Further, Bayesian non-parametrics allow the capacity of the model to grow with the data. Thereby we avoid imposing hard limits on the grammar (and the thorny problem of model selection), but instead allow the model to find a grammar appropriately sized for its training data.

4.1 Non-parametric form

Our Bayesian model of SCFG derivations resembles that of Blunsom et al. (2008). Given a grammar, each sentence is generated as follows. Starting with a root non-terminal (z_1), rewrite each frontier non-terminal (z_i) using a rule chosen from our grammar expanding z_i. Repeat until there are no remaining frontier non-terminals. This gives rise to the following derivation probability:

$$p(\mathbf{d}) = p(z_1) \prod_{r_i \in \mathbf{d}} p(r_i|z_i)$$

where the derivation is a sequence of rules $\mathbf{d} = (r_1, \ldots, r_n)$, and z_i denotes the root node of r_i.

We allow two types of rules: *non-terminal* and *terminal* expansions. The former rewrites a non-terminal symbol as a string of two or three non-terminals along with an alignment, specifying the corresponding ordering of the child trees in the source and target language. Terminal expansions rewrite a non-terminal as a pair of terminal n-grams, representing a phrasal translation pair, where either but not both may be empty.

Each rule in the grammar, r_i, is generated from its root symbol, z_i, by first choosing a rule type $t_i \in \{\texttt{TERM}, \texttt{NON-TERM}\}$ from a Bernoulli distribution, $r_i \sim \text{Bernoulli}(\gamma)$. We treat γ as a random variable with its own prior, $\gamma \sim \text{Beta}(\alpha^R, \alpha^R)$ and integrate out the parameters, γ. This results in the following conditional probability for t_i:

$$p(t_i|\mathbf{r}^{-i}, z_i, \alpha^R) = \frac{n^{-i}_{t_i,z_i} + \alpha^R}{n^{-i}_{\cdot,z_i} + 2\alpha^R}$$

where $n^{-i}_{r_i,z_i}$ is the number of times r_i has been used to rewrite z_i in the set of all other rules, $\mathbf{r}^{-i}$, and $n^{-i}_{\cdot,z_i} = \sum_r n^{-i}_{r,z_i}$ is the total count of rewriting z_i. The Dirichlet (and thus Beta) distribution are *exchangeable*, meaning that any permutation of its events are equiprobable. This allows us to reason about each event given previous and subsequent events (i.e., treat each item as the 'last'.)

When $t_i = \texttt{NON-TERM}$, we generate a binary or ternary non-terminal production. The non-terminal sequence and alignment are drawn from $(\mathbf{z}, a) \sim \phi^N_{z_i}$ and, as before, we define a prior over the parameters, $\phi^N_{z_i} \sim \text{Dirichlet}(\alpha^T)$, and integrate out $\phi^N_{z_i}$. This results in the conditional probability:

$$p(r_i|t_i = \texttt{NON-TERM}, \mathbf{r}^{-i}, z_i, \alpha^N) = \frac{n^{N,-i}_{r_i,z_i} + \alpha^N}{n^{N,-i}_{\cdot,z_i} + |N|\alpha^N}$$

where $n^{N,-i}_{r_i,z_i}$ is the count of rewriting z_i with non-terminal rule r_i, $n^{N,-i}_{\cdot,z_i}$ the total count over all non-terminal rules and $|N|$ is the number of unique non-terminal rules.

For terminal productions ($t_i = \texttt{TERM}$) we first decide whether to generate a phrase in both languages or in one language only, according to a fixed probability p_{null}.[3] Contingent on this decision, the terminal strings are then drawn from

[3] To discourage null alignments, we used $p_{null} = 10^{-10}$ for this value in the experiments we report below.

either $\phi^P_{z_i}$ for phrase pairs or ϕ^{null} for single language phrases. We choose Dirichlet process (DP) priors for these parameters:

$$\phi^P_{z_i} \sim \text{DP}(\alpha^P, P_1^P)$$
$$\phi^{null}_{z_i} \sim \text{DP}(\alpha^{null}, P_1^{null})$$

where the base distributions, P_1^P and P_1^{null}, range over phrase pairs or monolingual phrases in either language, respectively.

The most important choice for our model is the priors on the parameters of these terminal distributions. Phrasal SCFG models are subject to a degenerate maximum likelihood solution in which all probability mass is placed on long, or whole sentence, phrase translations (DeNero et al., 2006). Therefore, careful consideration must be given when specifying the P_1 distribution on terminals in order to counter this behavior.

To construct a prior over string pairs, first we define the probability of a monolingual string ($\mathbf{s}$):

$$P_0^X(\mathbf{s}) = P_{Poisson}(|\mathbf{s}|; 1) \times \frac{1}{V_X^{|\mathbf{s}|}}$$

where the $P_{Poisson}(k; 1)$ is the probability under a Poisson distribution of length k given an expected length of 1, while V_X is the vocabulary size of language X. This distribution has a strong bias towards short strings. In particular note that generally a string of length k will be less probable than two of length $\frac{k}{2}$, a property very useful for finding 'minimal' translation units. This contrasts with a geometric distribution in which a string of length k will be more probable than its segmentations.

We define P_1^{null} as the string probability of the non-$null$ part of the rule:

$$P_1^{null}(z \rightarrow \langle \mathbf{e}, \mathbf{f} \rangle) = \begin{cases} \frac{1}{2} P_0^E(\mathbf{e}) & \text{if } |\mathbf{f}| = 0 \\ \frac{1}{2} P_0^F(\mathbf{f}) & \text{if } |\mathbf{e}| = 0 \end{cases}$$

The terminal translation phrase pair distribution is a hierarchical Dirichlet Process in which each phrase are independently distributed according to DPs:[4]

$$P_1^P(z \rightarrow \langle \mathbf{e}, \mathbf{f} \rangle) = \phi_z^E(\mathbf{e}) \times \phi_z^F(\mathbf{f})$$
$$\phi_z^E \sim \text{DP}(\alpha^{P_E}, P_0^E)$$

[4]This prior is similar to one used by DeNero et al. (2008), who used the expected table count approximation presented in Goldwater et al. (2006). However, Goldwater et al. (2006) contains two major errors: omitting P_0, and using the truncated Taylor series expansion (Antoniak, 1974) which fails for small αP_0 values common in these models. In this work we track table counts directly.

and ϕ_z^F is defined analogously. This prior encourages frequent phrases to participate in many different translation pairs. Moreover, as longer strings are likely to be less frequent in the corpus this has a tendency to discourage long translation units.

4.2 A Gibbs sampler for derivations

Markov chain Monte Carlo sampling allows us to perform inference for the model described in 4.1 without restricting the infinite space of possible translation rules. To do this we need a method for sampling a derivation for a given sentence pair from $p(\mathbf{d}|\mathbf{d}^-)$. One possible approach would be to first build a packed chart representation of the derivation forest, calculate the inside probabilities of all cells in this chart, and then sample derivations top-down according to their inside probabilities (analogous to monolingual parse tree sampling described in Johnson et al. (2007)). A problem with this approach is that building the derivation forest would take $O(|\mathbf{f}|^3|\mathbf{e}|^3)$ time, which would be impractical for long sentences.

Instead we develop a collapsed Gibbs sampler (Teh et al., 2006) which draws new samples by making local changes to the derivations used in a previous sample. After a period of burn in, the derivations produced by the sampler will be drawn from the posterior distribution, $p(\mathbf{d}|\mathbf{x})$. The advantage of this algorithm is that we only store the current derivation for each training sentence pair (together these constitute the state of the sampler), but never need to reason over derivation forests. By integrating over (collapsing) the parameters we only store counts of rules used in the current sampled set of derivations, thereby avoiding explicitly representing the possibly infinite space of translation pairs.

We define two operators for our Gibbs sampler, each of which re-samples local derivation structures. Figures 2 and 4 illustrate the permutations these operators make to derivation trees. The omitted tree structure in these figures denotes the *Markov blanket* of the operator: the structure which is held constant when enumerating the possible outcomes for an operator.

The *Split/Join* operator iterates through the positions between each source word sampling whether a terminal boundary should exist at that position (Figure 2). If the source position

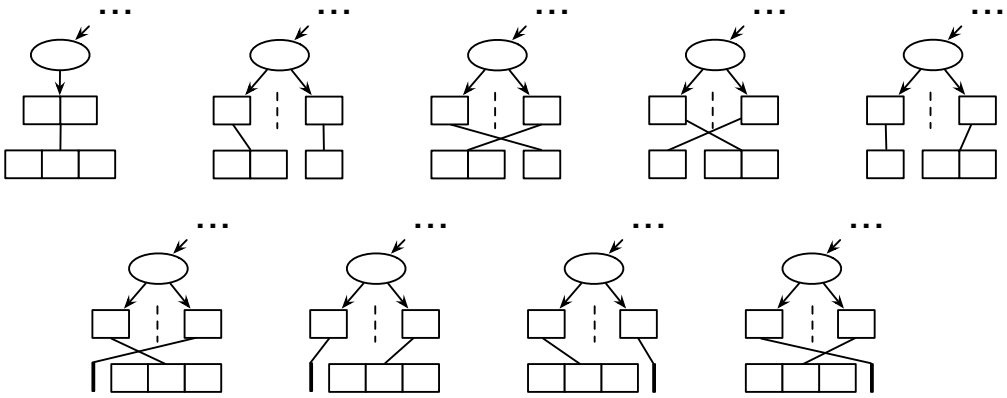


Figure 2: Split/Join sampler applied between a pair of adjacent terminals sharing the same parent. The dashed line indicates the source position being sampled, boxes indicate source and target tokens, while a solid line is a null alignment.

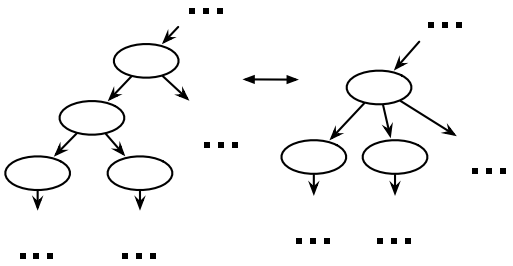


Figure 4: Rule insert/delete sampler. A pair of adjacent nodes in a ternary rule can be re-parented as a binary rule, or vice-versa.

falls between two existing terminals whose target phrases are adjacent, then any new target segmentation within those target phrases can be sampled, including $null$ alignments. If the two existing terminals also share the same parent, then any possible re-ordering is also a valid outcome, as is removing the terminal boundary to form a single phrase pair. Otherwise, if the visited boundary point falls within an existing terminal, then all target split and re-orderings are possible outcomes. The probability for each of these configurations is evaluated (see Figure 3) from which the new configuration is sampled.

While the first operator is theoretically capable of exploring the entire derivation forest (by flattening the tree into a single phrase and then splitting), the series of moves required would be highly improbable. To allow for faster mixing we employ the *Insert/Delete* operator which adds and deletes the parent non-terminal of a pair of adjacent nodes. This is illustrated in Figure 4. The update equations are analogous to those used for the Split/Join operator in Figure 3. In order for this operator to be effective we need to allow greater than binary branching nodes, otherwise deleting a nodes would require sampling from a much larger set of outcomes. Hence our adoption of a ternary branching grammar. Although such a grammar would be very inefficient for a dynamic programming algorithm, it allows our sampler to permute the internal structure of the trees more easily.

4.3 Hyperparameter Inference

Our model is parameterised by a vector of hyperparameters, $\alpha = (\alpha^R, \alpha^N, \alpha^P, \alpha^{P_E}, \alpha^{P_F}, \alpha^{null})$, which control the sparsity assumption over various model parameters. We could optimise each concentration parameter on the training corpus by hand, however this would be quite an onerous task. Instead we perform inference over the hyperparameters following Goldwater and Griffiths (2007) by defining a vague gamma prior on each concentration parameter, $\alpha^x \sim \text{Gamma}(10^{-4}, 10^4)$. This hyper-prior is relatively benign, allowing the model to consider a wide range of values for the hyperparameter. We sample a new value for each α^x using a log-normal distribution with mean α^x and variance 0.3, which is then accepted into the distribution $p(\alpha^x|\mathbf{d}, \alpha^-)$ using the Metropolis-Hastings algorithm. Unlike the Gibbs updates, this calculation cannot be distributed over a cluster (see Section 4.4) and thus is very costly. Therefore for small corpora we re-sample the hyperparameter after every pass through the corpus, for larger experiments we only re-sample every 20 passes.

4.4 A Distributed approximation

While employing a collapsed Gibbs sampler allows us to efficiently perform inference over the

$$p(\texttt{JOIN}) \propto p(t_i = \texttt{TERM}|z_i, \mathbf{r}^-) \times p(r_i = (z_i \rightarrow \langle \mathbf{e}, \mathbf{f} \rangle)|z_i, \mathbf{r}^-) \quad (1)$$

$$\begin{aligned} p(\texttt{SPLIT}) \propto\ & p(t_i = \texttt{NON-TERM}|z_i, \mathbf{r}^-) \times p(r_i = (z_i \rightarrow \langle z_l, z_r, a_i \rangle)|z_i, \mathbf{r}^-) \\ & \times p(t_l = \texttt{TERM}|t_i, z_i, \mathbf{r}^-) \times p(r_l = (z_l \rightarrow \langle \mathbf{e}_l, \mathbf{f}_l \rangle)|z_l, \mathbf{r}^-) \\ & \times p(t_r = \texttt{TERM}|t_i, t_l, z_i, \mathbf{r}^-) \times p(r_r = (z_r \rightarrow \langle \mathbf{e}_r, \mathbf{f}_r \rangle)|z_l, \mathbf{r}^- \cup (z_l \rightarrow \langle \mathbf{e}_l, \mathbf{f}_l \rangle)) \end{aligned} \quad (2)$$

Figure 3: Gibbs sampling equations for the competing configurations of the Split/Join sampler, shown in Figure 2. Eq. (1) corresponds to the top-left configuration, and (2) the remaining configurations where the choice of $\mathbf{e}_l, \mathbf{f}_l, \mathbf{e}_r, \mathbf{f}_r$ and a_i specifies the string segmentation and the alignment (monotone or reordered).

massive space of possible grammars, it induces dependencies between all the sentences in the training corpus. These dependencies make it difficult to scale our approach to larger corpora by distributing it across a number of processors. Recent work (Newman et al., 2007; Asuncion et al., 2008) suggests that good practical parallel performance can be achieved by having multiple processors independently sample disjoint subsets of the corpus. Each process maintains a set of rule counts for the entire corpus and communicates the changes it has made to its section of the corpus only after sampling every sentence in that section. In this way each process is sampling according to a slightly 'out-of-date' distribution. However, as we confirm in Section 5 the performance of this approximation closely follows the exact collapsed Gibbs sampler.

4.5 Extracting a translation model

Although we could use our model directly as a decoder to perform translation, its simple hierarchical reordering parameterisation is too weak to be effective in this mode. Instead we use our sampler to sample a distribution over translation models for state-of-the-art phrase based (Moses) and hierarchical (Hiero) decoders (Koehn et al., 2007; Chiang, 2007). Each sample from our model defines a hierarchical alignment on which we can apply the standard extraction heuristics of these models. By extracting from a sequence of samples we can directly infer a distribution over phrase tables or Hiero grammars.

5 Evaluation

Our evaluation aims to determine whether the phrase/SCFG rule distributions created by sampling from the model described in Section 4 impact upon the performance of state-of-the-art translation systems. We conduct experiments translating both Chinese (high reordering) and Arabic (low reordering) into English. We use the GIZA++ implementation of IBM Model 4 (Brown et al., 1993; Och and Ney, 2003) coupled with the phrase extraction heuristics of Koehn et al. (2003) and the SCFG rule extraction heuristics of Chiang (2007) as our benchmark. All the SCFG models employ a single X non-terminal, we leave experiments with multiple non-terminals to future work.

Our hypothesis is that our grammar based induction of translation units should benefit language pairs with significant reordering more than those with less. While for mostly monotone translation pairs, such as Arabic-English, the benchmark GIZA++-based system is well suited due to its strong monotone bias (the sequential Markov model and diagonal growing heuristic).

We conduct experiments on both small and large corpora to allow a range of alignment qualities and also to verify the effectiveness of our distributed approximation of the Bayesian inference. The samplers are initialised with trees created from GIZA++ Model 4 alignments, altered such that they are consistent with our ternary grammar. This is achieved by using the factorisation algorithm of Zhang et al. (2008a) to first create initial trees. Where these factored trees contain nodes with mixed terminals and non-terminals, or more than three non-terminals, we discard alignment points until the node factorises correctly. As the alignments contain many such non-factorisable nodes, these trees are of poor quality. However, all samplers used in these experiments are first 'burnt-in' for 1000 full passes through the data. This allows the sampler to diverge from its initialisation condition, and thus gives us confidence that subsequent samples will be drawn from the posterior. An expectation over phrase tables and Hiero grammars is built from every 50th sample after the burn-in, up until the 1500th sample.

We evaluate the translation models using IBM BLEU (Papineni et al., 2001). Table 1 lists the statistics of the corpora used in these experiments.

	IWSLT		NIST			
	English←Chinese		English←Chinese		English←Arabic	
Sentences	40k		300k		290k	
Segs./Words	380k	340k	11.0M	8.6M	9.3M	8.5M
Av. Sent. Len.	9	8	36	28	32	29
Longest Sent.	75	64	80	80	80	80

Table 1: Corpora statistics.

System	Test 05
Moses (Heuristic)	47.3
Moses (Bayes SCFG)	49.6
Hiero (Heuristic)	48.3
Hiero (Bayes SCFG)	51.8

Table 2: IWSLT Chinese to English translation.

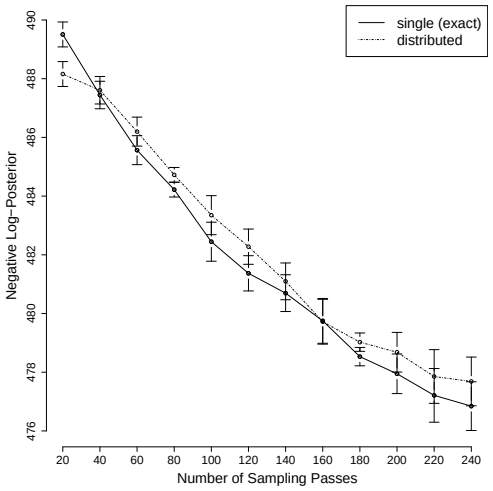


Figure 6: The posterior for the single CPU sampler and distributed approximation are roughly equivalent over a sampling run.

5.1 Small corpus

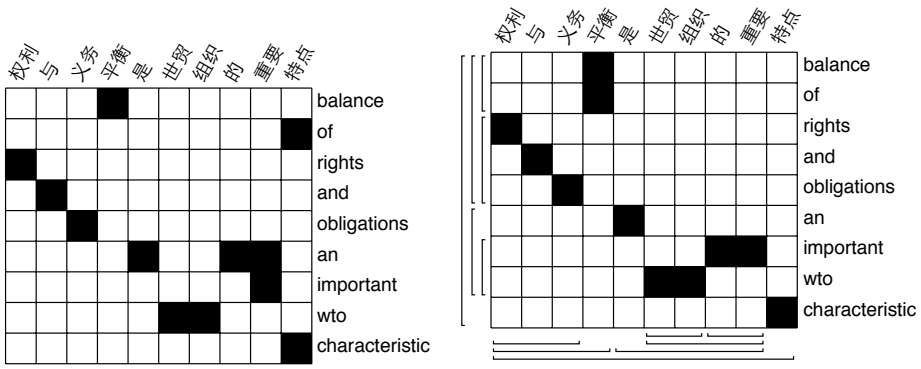
Firstly we evaluate models trained on a small Chinese-English corpus using a Gibbs sampler on a single CPU. This corpus consists of transcribed utterances made available for the IWSLT workshop (Eck and Hori, 2005). The sparse counts and high reordering for this corpus means the GIZA++ model produces very poor alignments.

Table 2 shows the results for the benchmark Moses and Hiero systems on this corpus using both the heuristic phrase estimation, and our proposed Bayesian SCFG model. We can see that our model has a slight advantage. When we look at the grammars extracted by the two models we note that the SCFG model creates considerably more translation rules. Normally this would suggest the alignments of the SCFG model are a lot sparser (more unaligned tokens) than those of the heuristic, however this is not the case. The projected SCFG derivations actually produce more alignment points. However these alignments are much more locally consistent, containing fewer spurious off-diagonal alignments, than the heuristic (see Figure 5), and thus produce far more valid phrases/rules.

5.2 Larger corpora

We now test our model's performance on a larger corpus, representing a realistic SMT experiment with millions of words and long sentences. The Chinese-English training data consists of the FBIS corpus (LDC2003E14) and the first 100k sentences from the Sinorama corpus (LDC2005E47). The Arabic-English training data consists of the eTIRR corpus (LDC2004E72), the Arabic news corpus (LDC2004T17), the Ummah corpus (LDC2004T18), and the sentences with confidence $c > 0.995$ in the ISI automatically extracted web parallel corpus (LDC2006T02). The Chinese text was segmented with a CRF-based Chinese segmenter optimized for MT (Chang et al., 2008). The Arabic text was preprocessed according to the D2 scheme of Habash and Sadat (2006), which was identified as optimal for corpora this size. The parameters of the NIST systems were tuned using Och's algorithm to maximize BLEU on the MT02 test set (Och, 2003).

To evaluate whether the approximate distributed inference algorithm described in Section 4.4 is effective, we compare the posterior probability of the training corpus when using a single machine, and when the inference is distributed on an eight core machine. Figure 6 plots the mean posterior and standard error for five independent runs for each scenario. Both sets of runs performed hyperparameter inference every twenty passes through the data. It is clear from the training curves that the distributed approximation tracks the corpus probability of the correct sampler sufficiently closely. This concurs with the findings of Newman et al.



(a) Giza++ (b) Gibbs

Figure 5: Alignment example. The synchronous tree structure is shown for (b) using brackets to indicate constituent spans; these are omitted for single token constituents. The right alignment is roughly correct, except that 'of' and 'an' should be left unaligned (是 'to be' is missing from the English translation).

System	MT03	MT04	MT05
Moses (Heuristic)	26.2	30.0	25.3
Moses (Bayes SCFG)	26.4	30.2	25.8
Hiero (Heuristic)	26.4	30.8	25.4
Hiero (Bayes SCFG)	26.7	30.9	26.0

Table 3: NIST Chinese to English translation.

System	MT03	MT04	MT05
Moses (Heuristic)	48.5	43.9	49.2
Moses (Bayes SCFG)	48.5	43.5	48.7
Hiero (Heuristic)	48.1	43.5	48.4
Hiero (Bayes SCFG)	48.4	43.4	47.7

Table 4: NIST Arabic to English translation.

(2007) who also observed very little empirical difference between the sampler and its distributed approximation.

Tables 3 and 4 show the result on the two NIST corpora when running the distributed sampler on a single 8-core machine.[5] These scores tally with our initial hypothesis: that the hierarchical structure of our model suits languages that exhibit less monotone reordering.

Figure 5 shows the projected alignment of a headline from the thousandth sample on the NIST Chinese data set. The effect of the grammar based alignment can clearly be seen. Where the combination of GIZA++ and the heuristics creates outlier alignments that impede rule extraction, the SCFG imposes a more rigid hierarchical structure on the alignments. We hypothesise that this property may be particularly useful for syntactic translation models which often have difficulty with inconsistent word alignments not corresponding to syntactic structure.

The combined evidence of the ability of our Gibbs sampler to improve posterior likelihood (Figure 6) and our translation experiments demonstrate that we have developed a scalable and effective method for performing inference over phrasal SCFG, without compromising the strong theoretical underpinnings of our model.

6 Discussion and Conclusion

We have presented a Bayesian model of SCFG induction capable of capturing phrasal units of translational equivalence. Our novel Gibbs sampler over synchronous derivation trees can efficiently draw samples from the posterior, overcoming the limitations of previous models when dealing with long sentences. This avoids explicitly representing the full derivation forest required by dynamic programming approaches, and thus we are able to perform inference without resorting to heuristic restrictions on the model.

Initial experiments suggest that this model performs well on languages for which the monotone bias of existing alignment and heuristic phrase extraction approaches fail. These results open the way for the development of more sophisticated models employing grammars capable of capturing a wide range of translation phenomena. In future we envision it will be possible to use the techniques developed here to directly induce grammars which match state-of-the-art decoders, such as Hiero grammars or tree substitution grammars of the form used by Galley et al. (2004).

[5] Producing the 1.5K samples for each experiment took approximately one day.

Acknowledgements

The authors acknowledge the support of the EPSRC (Blunsom & Osborne, grant EP/D074959/1; Cohn, grant GR/T04557/01) and the GALE program of the Defense Advanced Research Projects Agency, Contract No. HR0011-06-2-001 (Dyer).

References

C. E. Antoniak. 1974. Mixtures of dirichlet processes with applications to bayesian nonparametric problems. *The Annals of Statistics*, 2(6):1152–1174.

A. Asuncion, P. Smyth, M. Welling. 2008. Asynchronous distributed learning of topic models. In *NIPS*. MIT Press.

P. Blunsom, T. Cohn, M. Osborne. 2008. Bayesian synchronous grammar induction. In *Proceedings of NIPS 21*, Vancouver, Canada.

P. F. Brown, S. A. D. Pietra, V. J. D. Pietra, R. L. Mercer. 1993. The mathematics of statistical machine translation: Parameter estimation. *Computational Linguistics*, 19(2):263–311.

P.-C. Chang, D. Jurafsky, C. D. Manning. 2008. Optimizing Chinese word segmentation for machine translation performance. In *Proc. of the Third Workshop on Machine Translation*, Prague, Czech Republic.

C. Cherry, D. Lin. 2007. Inversion transduction grammar for joint phrasal translation modeling. In *Proc. of the HLT-NAACL Workshop on Syntax and Structure in Statistical Translation (SSST 2007)*, Rochester, USA.

D. Chiang. 2007. Hierarchical phrase-based translation. *Computational Linguistics*, 33(2):201–228.

J. DeNero, D. Klein. 2008. The complexity of phrase alignment problems. In *Proceedings of ACL-08: HLT, Short Papers*, 25–28, Columbus, Ohio. Association for Computational Linguistics.

J. DeNero, D. Gillick, J. Zhang, D. Klein. 2006. Why generative phrase models underperform surface heuristics. In *Proc. of the HLT-NAACL 2006 Workshop on Statistical Machine Translation*, 31–38, New York City.

J. DeNero, A. Bouchard-Côté, D. Klein. 2008. Sampling alignment structure under a Bayesian translation model. In *Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing*, 314–323, Honolulu, Hawaii. Association for Computational Linguistics.

M. Eck, C. Hori. 2005. Overview of the IWSLT 2005 evaluation campaign. In *Proc. of the International Workshop on Spoken Language Translation*, Pittsburgh.

M. Galley, M. Hopkins, K. Knight, D. Marcu. 2004. What's in a translation rule? In *Proc. of the 4th International Conference on Human Language Technology Research and 5th Annual Meeting of the NAACL (HLT-NAACL 2004)*, Boston, USA.

S. Goldwater, T. Griffiths. 2007. A fully bayesian approach to unsupervised part-of-speech tagging. In *Proc. of the 45th Annual Meeting of the ACL (ACL-2007)*, 744–751, Prague, Czech Republic.

S. Goldwater, T. Griffiths, M. Johnson. 2006. Contextual dependencies in unsupervised word segmentation. In *Proc. of the 44th Annual Meeting of the ACL and 21st International Conference on Computational Linguistics (COLING/ACL-2006)*, Sydney.

N. Habash, F. Sadat. 2006. Arabic preprocessing schemes for statistical machine translation. In *Proc. of the 6th International Conference on Human Language Technology Research and 7th Annual Meeting of the NAACL (HLT-NAACL 2006)*, New York City. Association for Computational Linguistics.

M. Johnson, T. Griffiths, S. Goldwater. 2007. Bayesian inference for PCFGs via Markov chain Monte Carlo. In *Proc. of the 7th International Conference on Human Language Technology Research and 8th Annual Meeting of the NAACL (HLT-NAACL 2007)*, 139–146, Rochester, New York.

P. Koehn, F. J. Och, D. Marcu. 2003. Statistical phrase-based translation. In *Proc. of the 3rd International Conference on Human Language Technology Research and 4th Annual Meeting of the NAACL (HLT-NAACL 2003)*, 81–88, Edmonton, Canada.

P. Koehn, H. Hoang, A. Birch, C. Callison-Burch, M. Federico, N. Bertoldi, B. Cowan, W. Shen, C. Moran, R. Zens, C. Dyer, O. Bojar, A. Constantin, E. Herbst. 2007. Moses: Open source toolkit for statistical machine translation. In *Proc. of the 45th Annual Meeting of the ACL (ACL-2007)*, Prague.

P. M. Lewis II, R. E. Stearns. 1968. Syntax-directed transduction. *J. ACM*, 15(3):465–488.

D. Marcu, W. Wong. 2002. A phrase-based, joint probability model for statistical machine translation. In *Proc. of the 2002 Conference on Empirical Methods in Natural Language Processing (EMNLP-2002)*, 133–139, Philadelphia. Association for Computational Linguistics.

D. Marcu, W. Wang, A. Echihabi, K. Knight. 2006. SPMT: Statistical machine translation with syntactified target language phrases. In *Proc. of the 2006 Conference on Empirical Methods in Natural Language Processing (EMNLP-2006)*, 44–52, Sydney, Australia.

D. Newman, A. Asuncion, P. Smyth, M. Welling. 2007. Distributed inference for latent dirichlet allocation. In *NIPS*. MIT Press.

F. J. Och, H. Ney. 2003. A systematic comparison of various statistical alignment models. *Computational Linguistics*, 29(1):19–52.

F. J. Och. 2003. Minimum error rate training in statistical machine translation. In *Proc. of the 41st Annual Meeting of the ACL (ACL-2003)*, 160–167, Sapporo, Japan.

K. Papineni, S. Roukos, T. Ward, W. Zhu. 2001. Bleu: a method for automatic evaluation of machine translation, 2001.

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

D. Wu. 1997. Stochastic inversion transduction grammars and bilingual parsing of parallel corpora. *Computational Linguistics*, 23(3):377–403.

H. Zhang, D. Gildea, D. Chiang. 2008a. Extracting synchronous grammar rules from word-level alignments in linear time. In *Proc. of the 22th International Conference on Computational Linguistics (COLING-2008)*, 1081–1088, Manchester, UK.

H. Zhang, C. Quirk, R. C. Moore, D. Gildea. 2008b. Bayesian learning of non-compositional phrases with synchronous parsing. In *Proc. of the 46th Annual Conference of the Association for Computational Linguistics: Human Language Technologies (ACL-08:HLT)*, 97–105, Columbus, Ohio.