An Infinite Hierarchical Bayesian Model of Phrasal Translation

Trevor Cohn
Department of Computer Science
The University of Sheffield
Sheffield, United Kingdom
t.cohn@sheffield.ac.uk

Gholamreza Haffari
Faculty of Information Technology
Monash University
Clayton, Australia
reza@monash.edu

Abstract

Modern phrase-based machine translation systems make extensive use of word-based translation models for inducing alignments from parallel corpora. This is problematic, as the systems are incapable of accurately modelling many translation phenomena that do not decompose into word-for-word translation. This paper presents a novel method for inducing phrase-based translation units directly from parallel data, which we frame as learning an inverse transduction grammar (ITG) using a recursive Bayesian prior. Overall this leads to a model which learns translations of entire sentences, while also learning their decomposition into smaller units (phrase-pairs) recursively, terminating at word translations. Our experiments on Arabic, Urdu and Farsi to English demonstrate improvements over competitive baseline systems.

1 Introduction

The phrase-based approach (Koehn et al., 2003) to machine translation (MT) has transformed MT from a narrow research topic into a truly useful technology to end users. Leading translation systems (Chiang, 2007; Koehn et al., 2007; Marcu et al., 2006) all use some kind of multi-word translation unit, which allows translations to be produced from large canned units of text from the training corpus. Larger phrases allow for the lexical context to be considered in choosing the translation, and also limit the number of reordering decisions required to produce a full translation.

Word-based translation models (Brown et al., 1993) remain central to phrase-based model training, where they are used to infer word-level alignments from sentence aligned parallel data, from which phrasal translation units are extracted using a heuristic. Although this approach demonstrably works, it suffers from a number of shortcomings. Firstly, many phrase-based phenomena which do not decompose into word translations (e.g., idioms) will be missed, as the underlying word-based alignment model is unlikely to propose the correct alignments. Secondly, the relationship between different phrase-pairs is not considered, such as between single word translations and larger multi-word phrase-pairs or where one large phrase-pair subsumes another.

This paper develops a phrase-based translation model which aims to address the above shortcomings of the phrase-based translation pipeline. Specifically, we formulate translation using inverse transduction grammar (ITG), and seek to learn an ITG from parallel corpora. The novelty of our approach is that we develop a Bayesian prior over the grammar, such that a nonterminal becomes a 'cache' learning each production *and* its complete yield, which in turn is recursively composed of its child constituents. This is closely related to adaptor grammars (Johnson et al., 2007a), which also generate full tree rewrites in a monolingual setting. Our model learns translations of entire sentences while also learning their decomposition into smaller units (phrase-pairs) recursively, terminating at word translations. The model is richly parameterised, such that it can describe phrase-based phenomena while also explicitly modelling the relationships between phrase-pairs and their component expansions, thus ameliorating the disconnect between the treatment of words versus phrases in the current MT pipeline. We develop a Bayesian approach using a Pitman-Yor process prior, which is capable of modelling a diverse range of geometrically decaying distributions over infinite event spaces (here translation phrase-pairs), an approach shown to be state of the art for language modelling (Teh, 2006).

Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics, pages 780–790,
Sofia, Bulgaria, August 4-9 2013.

We are not the first to consider this idea; Neubig et al. (2011) developed a similar approach for learning an ITG using a form of Pitman-Yor adaptor grammar. However Neubig et al.'s work was flawed in a number of respects, most notably in terms of their heuristic beam sampling algorithm which does not meet either of the Markov Chain Monte Carlo criteria of ergodicity or detailed balance. Consequently their approach does not constitute a valid Bayesian model. In contrast, this paper provides a more rigorous and theoretically sound method. Moreover our approach results in consistent translation improvements across a number of translation tasks compared to Neubig et al.'s method, and a competitive phrase-based baseline.

2 Related Work

Inversion transduction grammar (or ITG) (Wu, 1997) is a well studied synchronous grammar formalism. Terminal productions of the form $X \rightarrow e/f$ generate a word in two languages, and non-terminal productions allow phrasal movement in the translation process. Straight productions, denoted by their non-terminals inside square brackets [...], generate their symbols in the given order in both languages, while inverted productions, indicated by angled brackets $\langle ... \rangle$, generate their symbols in the reverse order in the target language.

In the context of machine translation, ITG has been explored for statistical word alignment in both unsupervised (Zhang and Gildea, 2005; Cherry and Lin, 2007; Zhang et al., 2008; Pauls et al., 2010) and supervised (Haghighi et al., 2009; Cherry and Lin, 2006) settings, and for decoding (Petrov et al., 2008). Our paper fits into the recent line of work for jointly inducing the phrase table and word alignment (DeNero and Klein, 2010; Neubig et al., 2011). The work of DeNero and Klein (2010) presents a *supervised* approach to this problem, whereas our work is *unsupervised* hence more closely related to Neubig et al. (2011) which we describe in detail below.

A number of other approaches have been developed for learning phrase-based models from bilingual data, starting with Marcu and Wong (2002) who developed an extension to IBM model 1 to handle multi-word units. This pioneering approach suffered from intractable inference and moreover, suffers from degenerate solutions (DeNero and Klein, 2010). Our approach is similar to these previous works, except that we impose additional constraints on how phrase-pairs can be tiled to produce a sentence pair, and moreover, we seek to model the embedding of phrase-pairs in one another, something not considered by this prior work. Another strand of related research is in estimating a broader class of synchronous grammars than ITGs, such as SCFGs (Blunsom et al., 2009b; Levenberg et al., 2012). Conceptually, our work could be readily adapted to general SCFGs using similar techniques.

This work was inspired by adaptor grammars (Johnson et al., 2007a), a monolingual grammar formalism whereby a non-terminal rewrites in a single step as a complete subtree. The model prior allows for trees to be generated as a mixture of a cache and a base adaptor grammar. In our case, we have generalised to a bilingual setting using an ITG. Additionally, we have extended the model to allow recursive nesting of adapted non-terminals, such that we end up with an infinitely recursive formulation where the top-level and base distributions are explicitly linked together.

As mentioned above, ours is not the first work attempting to generalise adaptor grammars for machine translation; (Neubig et al., 2011) also developed a similar approach based around ITG using a Pitman-Yor Process prior. Our approach improves upon theirs in terms of the model and inference, and critically, this is borne out in our experiments where we show uniform improvements in translation quality over a baseline system, as compared to their almost entirely negative results. We believe that their approach had a number of flaws: For inference they use a beam-search, which may speed up processing but means that they are no longer sampling from the true distribution, nor a distribution with the same support as the posterior. Moreover they include a Metropolis-Hastings correction step, which is required to correct the samples to account for repeated substructures which will be otherwise underrepresented. Consequently their approach does not constitute a Markov Chain Monte Carlo sampler, but rather a complex heuristic.

The other respect in which this work differs from Neubig et al. (2011) is in terms of model formulation. They develop an ITG which generates phrase-pairs as terminals, while we employ a more restrictive word-based model which forces the decomposition of every phrase-pair. This is an important restriction as it means that we jointly learn

a word and phrase based model, such that word based phenomena can affect the phrasal structures. Finally our approach models separately the three different types of ITG production (monotone, swap and lexical emission), allowing for a richer parameterisation which the model exploits by learning different hyper-parameter values.

3 Model

The generative process of the model follows that of ITG with the following simple grammar

$$X \to [X\ X] \mid \langle X\ X \rangle$$
$$X \to e/f \mid e/\bot \mid \bot/f\,,$$

where $[\cdot]$ denotes monotone ordering and $\langle\cdot\rangle$ denotes a swap in one language. The symbol $\bot$ denotes the empty string. This corresponds to a simple generative story, with each stage being a non-terminal rewrite starting with X and terminating when there are no frontier non-terminals.

A popular variant is a *phrasal ITG*, where the leaves of the ITG tree are phrase-pairs and the training seeks to learn a segmentation of the source and target which yields good phrases. We would not expect this model to do very well as it cannot consider overlapping phrases, but instead is forced into selecting between many competing – and often equally viable – options. Our approach improves over the phrasal model by recursively generating complete phrases. This way we don't insist on a single tiling of phrases for a sentence pair, but explicitly model the set of hierarchically nested phrases as defined by an ITG derivation. This approach is closer in spirit to the phrase-extraction heuristic, which defines a set of 'atomic' terminal phrase-pairs and then extracts every combination of these atomic phase-pairs which is contiguous in the source and target.[1]

The generative process is that we draw a complete ITG tree, $t \sim P_2(\cdot)$, as follows:

1. choose the rule type, $r \sim R$, where $r \in \{\texttt{mono}, \texttt{swap}, \texttt{emit}\}$
2. for $r = \texttt{mono}$
   (a) draw the complete subtree expansion, $t = X \to [\ldots] \sim T_M$
3. for $r = \texttt{swap}$
   (a) draw the complete subtree expansion, $t = X \to \langle\ldots\rangle \sim T_S$
4. for $r = \texttt{emit}$
   (a) draw a pair of strings, $(e, f) \sim E$
   (b) set $t = X \to e/f$

Note that we split the problem of drawing a tree into two steps: first choosing the top-level rule type and then drawing a rule of that type. This gives us greater control than simply drawing a tree of any type from one distribution, due to our parameterisation of the priors over the model parameters T_M, T_S and E.

To complete the generative story, we need to specify the prior distributions for T_M, T_S and E. First, we deal with the emission distribution, E which we drawn from a Dirichlet Process prior $E \sim \text{DP}(b_E, P_0)$. We restrict the emission rules to generate word pairs rather than phrase pairs.[2] For the base distribution, P_0, we use a simple uniform distribution over word pairs,

$$P_0(e,f) = \begin{cases} \eta^2 \frac{1}{V_E V_F} & e \neq \bot,\ f \neq \bot \\ \eta(1-\eta)\frac{1}{V_F} & e = \bot, f \neq \bot \\ \eta(1-\eta)\frac{1}{V_E} & e \neq \bot, f = \bot \end{cases},$$

where the constant η denotes the binomial probability of a word being aligned.[3]

We use Pitman-Yor Process priors for the T_M and T_S parameters

$$T_M \sim \text{PYP}(a_M, b_M, P_1(\cdot|r = \texttt{mono}))$$
$$T_S \sim \text{PYP}(a_S, b_S, P_1(\cdot|r = \texttt{swap}))$$

where $P_1(t_1, t_2|r)$ is a distribution over a pair of trees (the left and right children of a monotone or swap production). P_1 is defined as follows:

1. choose the complete left subtree $t_1 \sim P_2$,
2. choose the complete right subtree $t_2 \sim P_2$,
3. set $t = X \to [t_1\ t_2]$ or $t = X \to \langle t_1\ t_2\rangle$ depending on r

This generative process is mutually recursive: P_2 makes draws from P_1 and P_1 makes draws from P_2. The recursion is terminated when the rule type $r = \texttt{emit}$ is drawn.

Following standard practice in Bayesian models, we integrate out R, T_M, T_S and E. This means draws from P_2 (or P_1) are no longer *iid*: for any non-trivial tree, computing its probability under this model is complicated by the fact

[1] Our technique considers the subset of phrase-pairs which are consistent with the ITG tree.

[2] Note that we could allow phrases here, but given the model can already reason over phrases by way of its hierarchical formulation, this is an unnecessary complication.

[3] We also experimented with using word translation probabilities from IBM model 1, based on the prior used by Levenberg et al. (2012), however we found little empirical difference compared with this simpler uniform model.

that the probability of its two subtrees are interdependent. This is best understood in terms of the Chinese Restaurant Franchise (CRF; Teh et al. (2006)), which describes the posterior distribution after integrating out the model parameters. In our case we can consider the process of drawing a tree from P_2 as a customer entering a restaurant and choosing where to sit, from an infinite set of tables. The seating decision is based on the number of other customers at each table, such that popular tables are more likely to be joined than unpopular or empty ones. If the customer chooses an occupied table, the identity of the tree is then set to be the same as for the other customers also seated there. For empty tables the tree must be sampled from the base distribution P_1. In the standard CRF analogy, this leads to another customer entering the restaurant one step up in the hierarchy, and this process can be chained many times. In our case, however, every new table leads to new customers reentering the original restaurant – these correspond to the left and right child trees of a monotone or swap rule. The recursion terminates when a table is shared, or a new table is labelled with a emit rule.

3.1 Inference

The probability of a tree (i.e., a draw from P_2) under the model is

$$P_2(t) = P(r)P_2(t|r) \tag{1}$$

where r is the rule type, one of `mono`, `swap` or `emit`. The distribution over types, $P(r)$, is defined as

$$P(r) = \frac{n_r^{T,-} + b_T \frac{1}{3}}{n^{T,-} + b_T}$$

where $n^{T,-}$ are the counts over rules of types.[4]

The second component in (1), $P_2(t|r)$, is defined separately for each rule type. For $r = \texttt{mono}$ or $r = \texttt{swap}$ rules, it is defined as

$$P_2(t|r) = \frac{n_{t,r}^- - K_{t,r}^- a_r}{n_r^- + b_r} + \frac{K_r^- a_r + b_r}{n_r^- + b_r} P_1(t_1, t_2|r)\,, \tag{2}$$

where $n_{t,r}^-$ is the count for tree t in the other training sentences, $K_{t,r}^-$ is the table count for t and n_r^- and K_r^- are the total count of trees and tables, respectively. Finally, the probability for $r = \texttt{emit}$ is given by

$$P_2(t|r = \texttt{emit}) = \frac{n_{t,E}^- + b_E P_0(e,f)}{n_r^- + b_r}\,,$$

where $t = X \to e/f$.

To complete the derivation we still need to define P_1, which is formulated as

$$P_1(t_1, t_2) = P_2(t_1)P_2(t_2|t_1)\,,$$

where the conditioning of the second recursive call to P_2 reflects that the counts n^- and K^- may be affected by the first draw from P_2. Although these two draws are assumed *iid* in the prior, after marginalising out T they are no longer independent. For this reason, evaluating $P_2(t)$ is computationally expensive, requiring tracking of repeated substructures in descendent sub-trees of t, which may affect other descendants. This results in an asymptotic complexity exponential in the number of nodes in the tree. For this reason we consider trees annotated with binary values denoting their table assignment, namely whether they share a table or are seated alone. Given this, the calculation is greatly simplified, and has linear complexity.[5]

We construct an approximating ITG following the technique used for sampling trees from monolingual tree-substitution grammars (Cohn et al., 2010). To do so we encode the first term from (2) separately from the second term (corresponding to draws from P_1). Summing together these two alternate paths – i.e., during inside inference – we recover P_2 as shown in (2). The full grammar transform for inside inference is shown in Table 1.

The sampling algorithm closely follows the process for sampling derivations from Bayesian PCFGs (Johnson et al., 2007b). For each sentence-pair, we first decrement the counts associated with its current tree, and then sample a new derivation. This involves first constructing the inside lattice using the productions in Table 1, and then performing a top-down sampling pass. After sampling each derivation from the approximating grammar, we then convert this into its corresponding ITG tree, which we then score with the full model and accept or reject the sample using the

[4] The conditioning on event and table counts, n^-, K^- is omitted for clarity.

[5] To support this computation, we track explicit table assignments for every training tree and their component subtrees. We also sample trees labelled with seating indicator variables.

Type	$X \to M$	$P(r = \texttt{mono})$
	$X \to S$	$P(r = \texttt{swap})$
	$X \to E$	$P(r = \texttt{emit})$
Base	$M \to [XX]$	$\frac{K_M^- a_M + b_M}{n_M^- + b_M}$
	$S \to \langle XX \rangle$	$\frac{K_S^- a_S + b_S}{n_S^- + b_S}$
Count	For every tree, t, of type $r = \texttt{mono}$, with $n_{t,M} > 0$:	
	$M \to \text{sig}(t)$	$\frac{n_{t,M}^- - K_{t,M}^- a_r}{n_M^- + b_M}$
	$\text{sig}(t) \to \text{yield}(t)$	1
	For every tree, t, of type $r = \texttt{swap}$, with $n_{t,S} > 0$:	
	$S \to \text{sig}(t)$	$\frac{n_{t,S}^- - K_{t,S}^- a_S}{n_S^- + b_S}$
	$\text{sig}(t) \to \text{yield}(t)$	1
Emit	For every word pair, e/f in sentence pair, where one of e, f can be $\perp$:	
	$E \to e/f$	$P_2(t)$

Table 1: Grammar transformation rules for MAP inside inference. The function sig(t) returns a unique identifier for the complete tree t, and the function $yield(t)$ returns the pair of terminal strings from the yield of t.

Metropolis-Hastings algorithm.[6] Accepted samples then replace the old tree (otherwise the old tree is retained) and the model counts are incremented. This process is then repeated for each sentence pair in the corpus in a random order.

4 Experiments

Datasets We train our model across three language pairs: Urdu→English (UR-EN), Farsi→English (FA-EN), and Arabic→English (AR-EN). The corpora statistics of these translation tasks are summarised in Table 2. The UR-EN corpus comes from NIST 2009 translation evaluation.[7] The AR-EN 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). For FA-EN, we use TEP[8] Tehran English-Persian Parallel corpus (Pilevar and Faili, 2011), which consists of conversational/informal text extracted from 1600 movie subtitles. We tokenized this corpus, removed noisy single-word sentences, randomly selected the development and test sets, and used the rest of the corpus as the training set. We discard sentences with length above 30 from the datasets for all experiments.[9]

	source	target	sentences
UR-EN	745K	575K	148K
FA-EN	4.7M	4.4M	498K
AR-EN	1.94M	2.08M	113K

Table 2: Corpora statistics showing numbers of parallel sentences and source and target words for the training sets.

Sampler configuration Samplers are initialised with trees created from GIZA++ alignments constructed using a SCFG *factorisation* method (Blunsom et al., 2009a). This algorithm represents the translation of a sentence as a large SCFG rule, which it then factorises into lower rank SCFG rules, a process akin to rule binarisation commonly used in SCFG decoding. Rules that cannot be reduced to a rank-2 SCFG are simplified by dropping alignment edges until they can be factorised, the net result being an ITG derivation largely respecting the alignments.[10]

The blocked sampler was run 1000 iterations for UR-EN, 100 iterations for FA-EN and AR-EN. After each full sampling iteration, we resample all the hyper-parameters using slice-sampling, with the following priors: $a \sim \text{Beta}(1, 1)$, $b \sim \text{Gamma}(10, 0.1)$. Figure 1 shows the posterior probability improves with each full sampling iterations. The alignment probability was set to $\eta = 0.99$. The sampling was repeated for 5 independent runs, and we present results where we combine the outputs of these runs. This is a form of Monte Carlo integration which allows us to represent the uncertainty in the posterior, while also representing multiple modes, if present.

The time complexity of our inference algorithm is $O(n^6)$, which can be prohibitive for large scale machine translation tasks. We reduce the complexity by constraining the *inside* inference to consider only derivations which are compatible

[6] The full model differs from the approximating grammar in that it accounts for inter-dependencies between subtrees by recursively tracking the changes in the customer and table counts while scoring the tree. Around 98% of samples were accepted in our experiments.

[7] http://www.itl.nist.gov/iad/mig/tests/mt/2009

[8] http://ece.ut.ac.ir/NLP/resources.htm

[9] Hence the BLEU scores we get for the baselines may appear lower than what reported in the literature.

[10] Using the factorised alignments directly in a translation system resulted in a slight loss in BLEU versus using the unfactorised alignments. Our baseline system uses the latter.

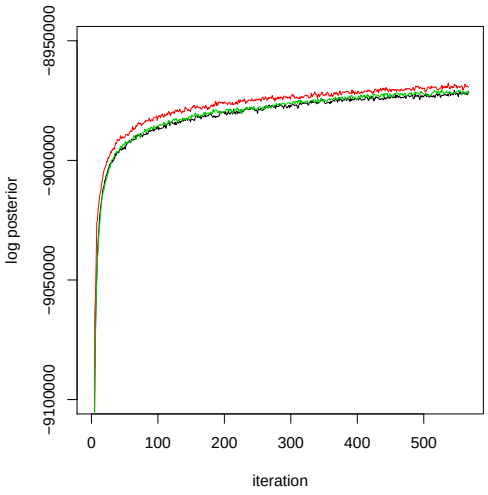


Figure 1: Training progress on the UR-EN corpus, showing the posterior probability improving with each full sampling iteration. Different colours denote independent sampling runs.

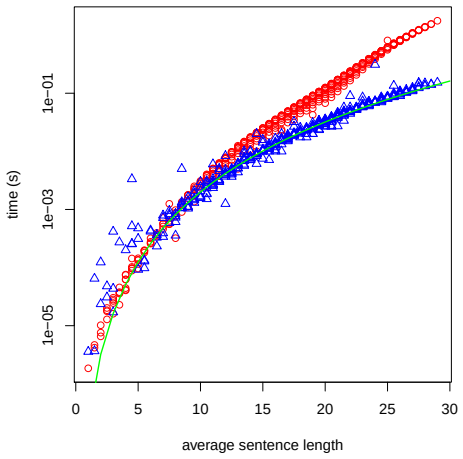


Figure 2: The runtime cost of bottom-up inside inference and top-down sampling as a function of sentence length (UR-EN), with time shown on a logarithmic scale. Full ITG inference is shown with red circles, and restricted inference using the intersection constraints with blue triangles. The average time complexity for the latter is roughly $O(l^4)$, as plotted in green $t = 2 \times 10^{-7} l^4$.

with high confidence alignments from GIZA++.[11] Figure 2 shows the sampling time with respect to the average sentence length, showing that our alignment-constrained sampling algorithm is better than the unconstrained algorithm with empirical complexity of n^4. However, the time complexity is still high, so we set the maximum sentence length to 30 to keep our experiments practicable. Presumably other means of inference may be more efficient, such as Gibbs sampling (Levenberg et al., 2012) or auxiliary variable sampling (Blunsom and Cohn, 2010); we leave these extensions to future work.

Baselines. Following (Levenberg et al., 2012; Neubig et al., 2011), we evaluate our model by using its output word alignments to construct a phrase table. As a baseline, we train a phrase-based model using the moses toolkit[12] based on the word alignments obtained using GIZA++ in both directions and symmetrized using the grow-diag-final-and heuristic[13] (Koehn et al., 2003). This alignment is used as input to the rule factorisation algorithm, producing the ITG trees with which we initialise our sampler. To put our results in the context of the previous work, we also compare against *pialign* (Neubig et al., 2011), an ITG algorithm using a Pitman-Yor process prior, as described in Section 2.[14]

In the end-to-end MT pipeline we use a standard set of features: relative-frequency and lexical translation model probabilities in both directions; distance-based distortion model; language model and word count. We set the distortion limit to 6 and max-phrase-length to 7 in all experiments. We train 3-gram language models using modified Kneser-Ney smoothing. For AR-EN experiments the language model is trained on English data as (Blunsom et al., 2009a), and for FA-EN and UR-EN the English data are the target sides of the bilingual training data. We use minimum error rate training (Och, 2003) with nbest list size 100 to optimize the feature weights for maximum development BLEU.

[11] These are taken from the final model 4 word alignments, using the intersection of the source-target and target-source models. These alignments are very high precision (but have low recall), and therefore are unlikely to harm the model.

[12] http://www.statmt.org/moses

[13] We use the default parameter settings in both moses and GIZA++.

[14] http://www.phontron.com/pialign

		Baselines		This paper	
		GIZA++	pialign	individual	combination
UR-EN		16.95	15.65	16.68 ± .12	**16.97**
FA-EN		20.69	21.41	21.36 ± .17	**21.50**
AR-EN	MT03	44.05	43.30	44.8 ± .28	**45.10**
	MT04	38.15	37.78	38.4 ± .08	**38.4**
	MT05	42.81	42.18	43.13 ± .23	**43.45**
	MT08	32.43	**33.00**	32.7 ± .15	32.80

Table 3: The BLEU scores for the translation tasks of three language pairs. The individual *column show the average and 95% confidence intervals for 5 independent runs, whereas the* combination *column show the results for combining the phrase tables of all these runs. The baselines are* GIZA++ *alignments and those generated by the pialign (Neubig et al., 2011)* **bold:** *the best result.*

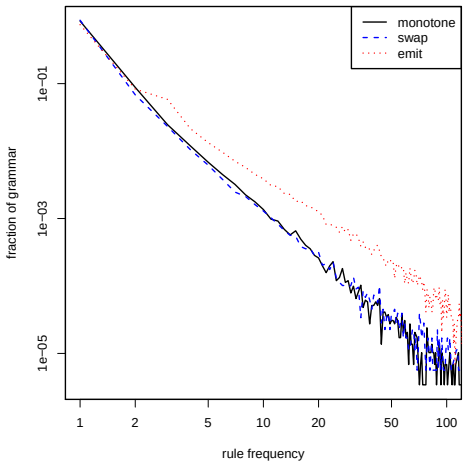


Figure 3: Fraction of rules with a given frequency, using a single sample grammar (UR-EN).

4.1 Results

Table 3 shows the BLEU scores for the three translation tasks UR/AR/FA→EN based on our method against the baselines. For our models, we report the average BLEU score of the 5 independent runs as well as that of the *aggregate* phrase table generated by these 5 independent runs. There are a number of interesting observations in Table 3. Firstly, combining the phrase tables from independent runs results in increased BLEU scores, possibly due to the representation of uncertainty in the outputs, and the representation of different modes captured by the individual models. We believe this type of Monte Carlo model averaging should be considered in general when sampling techniques are employed for grammatical inference, e.g. in parsing and translation. Secondly, our approach consistently improves over the Giza++ baseline often by a large margin, whereas *pialign* underperforms the GIZA++ baseline in many cases. Thirdly, our model consistently outperforms *pialign* (except in AR-EN MT08 which is very close). This highlights the modeling and inference differences between our method and the *pialign*.

5 Analysis

In this section, we present some insights about the learned grammar and the model hyper-parameters. Firstly, we start by presenting various statistics about different learned grammars. Figure 3 shows the fraction of rules with a given frequency for each of the three rule types. The three types of rule exhibit differing amounts of high versus low frequency rules, and all roughly follow power laws. As expected, there is a higher tendency to reuse high-frequency emissions (or single-word translation) compared to other rule types, which are the basic building blocks to compose larger rules (or phrases). Table 4 lists the high frequency monotone and swap rules in the learned grammar. We observe the high frequency swap rules capture reordering in verb clusters, preposition-noun inversions and adjective-noun reordering. Similar patterns are seen in the monotone rules, along with some common canned phrases. Note that "in Iraq" appears twice, once as an inversion in UR-EN and another time in monotone order for AR-EN.

Secondly, we analyse the values learned for the model hyper-parameters; Figure 4.(a) shows the posterior distribution over the hyper-parameter values. There is very little spread in the inferred values, suggesting the sampling chains may have converged. Furthermore, there is a large difference between the learned hyper-parameters for the monotone rules versus the swap rules. For the Pitman-Yor Process prior, the values of the hyper-

	FARSI-ENGLISH
pialign	sure/mTm n,chief/r ys, you sure/mTm ny, im sure/mTm nm, boss/r ys, make sure/mTm n, are you sure/mTm ny, anyway/hr HAl, president/r ys jmhwr, not sure/mTm n nystm, im sure/mTm nm kh, i'm sure/mTm nm, sure/mTm nA
our method	sure/mTm }, have/dA $ th, be/b $, have/dA $ th bA $, let me/* Ar, because of/xATr, sure/mTm } n, do/kAr rA, come on/zwd bA, excuse me/bbx, kill/rA bk, come on/zwdbA, more than/$ tr, behind/p $, what do/mnZwrt, what do you/mnZwrt , kill/k $, dont worry/ngrAn nbA, is it/$ dh, welcome/xw $ |, chief/r } ys, make sure/mTm, is/my $, make sure/mTm }, make sure/mTm} n, im sorry/bbx, left/g * A, if/Agr $
	ARABIC-ENGLISH
pialign	said ./و- قال, states/المتحدة, united/الولايات, al-wafd/1 ص الوفد, efforts/بذل, of mass destruction/الدمار الشامل, youm/اليوم, jintao/جينتاو, alam al youm/العالم اليوم, al-ittihad ,/الاتحاد, the field of/مجال, islamabad/اسلام, scheduled/من المقرر, al-alam al-youm/العالم اليوم ص, / al-hayat/الحياة, peninsula/شبه الجزيرة, meanwhile/الوقت نفسه, prime/وزراء, the invitation/دعوة من, / al-hayat ,/الحياة, korean peninsula/شبه الجزيرة الكورية, al-nahar "/النهار ``, department/وزارة الخارجية, cote/لكوت, as possible/وقت ممكن, al alam al youm/العالم اليوم , al-alam al-youm ,/العالم اليوم, at the invitation/ب- دعوة من, jacques/جاك الفرنسي, well as/لكذا, points/يلي, vladimir putin/الروسي فلاديمير بوتين, george w. bush/جورج
our method	the united/المتحدة, us dollars/امريكي, prime/رئيس الوزراء, china '/الصين, spokesman/المتحدث -ب, many/من, is expected/المتوقع, is expected to/المتوقع, at least/الاقل, on tuesday/يوم, egypt '/مصر, thursday/يوم, the un/المتحدة, on thursday/يوم, friday/يوم, on friday/يوم, to/حسب, al-wafd ,/الوفد ص, the us/المتحدة, for/-ل النسبة, first time/المرة الاولى, further/من, iraq '/العراق, israeli prime/رئيس الوزراء الاسرائيلي, the two/البلدين, on saturday/يوم, on sunday/يوم, u.s./المتحدة, views/النظر, sharon '/شارون, country '/البلد, he said/ذكر, israel '/اسرائيل, people '/الشعب, here/هنا اليوم, china '/للصين, , he said/و- اضاف, earlier/فى وقت, china '/الصينية, at least/ما لا يقل, the u.s./المتحدة, the gaza/قطاع, the gaza strip/قطاع, he added/قال, are expected/المتوقع, are expected to/المتوقع, are expected to/من المتوقع, million u.s./مليون, according/قال, to/تواصل, order/من, in order/من, he pointed/اشار, mfa , asharq al/الاوسط ص, mfa , asharq al awsat/الاوسط ص, arafat '/عرفات

Table 5: Good phrase pairs in the top-100 high frequency phrase pairs specific to the phrase tables coming from our method vs that of pialign for FA-EN and AR-EN translation tasks.

parameters affects the rate at which the number of types grows compared to the number of tokens. Specifically, as the discount a or the concentration b parameters increases we expect for a relative increase in the number of types. If the number of observed monotone and swap rules were equal, then there would be a higher chance in reusing the monotone rules. However, the number of observed monotone and swap rules are not equal, as plotted in Figure 4.(b). Similar results were observed for the other language pairs (figures omitted for space reasons).

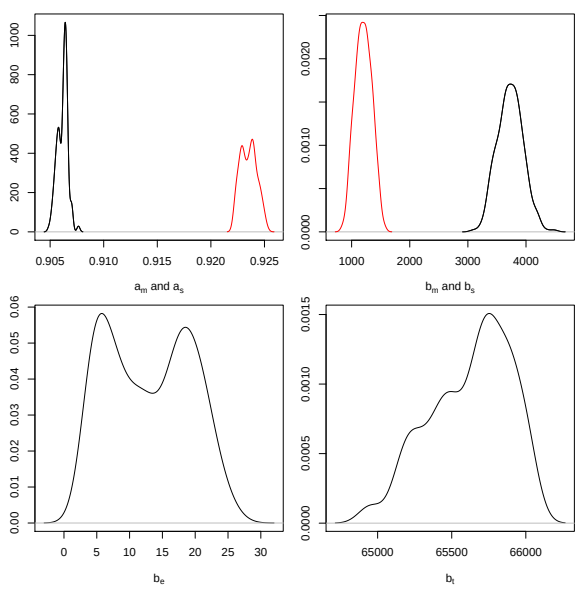
Thirdly, we performed a manual evaluation for the quality of the phrase-pairs learned exclusively by our method vs *pialign*. For each method, we considered the top-100 high frequency phrase-pairs which are specific to that method. Then we asked a bilingual human expert to identify reasonably well phrase-pairs among these top-100 phrase-pairs. The results are summarized in Table 5, and show that we learn roughly twice as many reasonably good phrase-pairs for AR-EN and FA-EN compared to *pialign*.

Conclusions

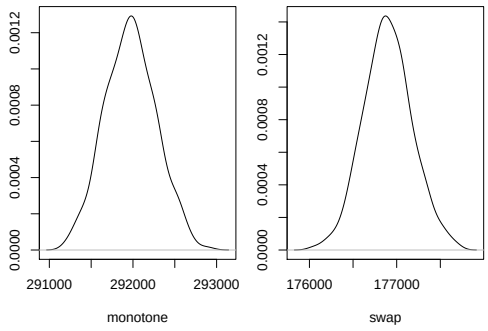
We have presented a novel method for learning a phrase-based model of translation directly from parallel data which we have framed as learning an inverse transduction grammar (ITG) using a recursive Bayesian prior. This has led to a model which learns translations of entire sentences, while also learning their decomposition into smaller units (phrase-pairs) recursively, terminating at word translations. We have presented a Metropolis-Hastings sampling algorithm for blocked inference in our non-parametric ITG. Our experiments on Urdu-English, Arabic-English, and Farsi-English translation tasks all demonstrate improvements over competitive baseline systems.

Acknowledgements

The first author was supported by the EPSRC (grant EP/I034750/1) and an Erasmus-Mundus scholarship funding a research visit to Melbourne. The second author was supported by an early career research award from Monash University.



(a)



(b)

Figure 4: (a) Posterior over the hyper-parameters, $a_M, a_S, b_M, b_S, b_E, b_T$, measured for UR-EN using samples 400–500 for 3 independent sampling chains, and the intersection constraints. (b) Posterior over the number of monotone and swap rules in the resultant grammars. The distribution for emission rules was also peaked about 147k rules.

260	[[[he/AnhwN] [⊥/nY]] [said/khA]]
222	[⟨ [said/khA] [⊥/nY] ⟩ [that/kh]]
219	[[[[he/AnhwN] [⊥/nY]] [said/khA]] [that/kh]]
148	[[[if/Agr] [⊥/|]] [you/p]]
108	[[he/AnhwN] ⟨ [said/khA] [⊥/nY] ⟩]
182	⟨ [will/gA] [be/hw] ⟩
129	⟨ [is/hY] [not/nhyN] ⟩
123	⟨ [has/hY] [been/gyA] ⟩
104	⟨ [will/gA] [be/jAyY] ⟩
103	⟨ [in/myN] [iraq/ErAq] ⟩
	Urdu-English
890	[[one/yky] [of/Az]]
843	[⟨ [yeah/rh] [⊥/|] ⟩ [./.]]
738	[[with/bA] [me/mn]]
644	[[[[okay/bA] [⊥/$]] [⊥/h]] [./.]]
608	[[to/bh] [me/mn]]
251	⟨ [is/dh] [it/$] ⟩
220	⟨ [tell/bgw] [me/mn] ⟩
199	⟨ < [i/⊥] [can/twnm] ⟩ ['t/nmy] >
190	⟨ [[who/ky] [are/hsty]] [you/tw] ⟩
187	⟨ [told/gft] [me/mn] ⟩
	Farsi-English
566	[[in/فی] [iraq/الع‌راق]]
414	[[in/في] [egypt/مصر]]
391	[[this/هذا] [year/الع‌ام]]
356	[[asharq/الش‌رق] [al-awsat/الاوسط]]
300	[[in/في] [iraq/الع‌راق]]
4024	⟨ [./.] ["/"] ⟩
1312	⟨ < [the/] [united/الم‌ت‌حدة] ⟩ [states/الولايات] >
665	⟨ [united/الم‌ت‌حدة] [states/الولايات] ⟩
650	⟨ [last/الماضي] [year/الع‌ام] ⟩
467	⟨ < [the/] [united/الم‌ت‌حدة] ⟩ [nations/الامم] >
	Arabic-English

Table 4: Top 5 monotone and swap productions and their counts. Rules with mostly punctuation or encoding 1:many or many:1 alignments were omitted.

References

Phil Blunsom and Trevor Cohn. 2010. Inducing synchronous grammars with slice sampling. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 238–241, Los Angeles, California, June. Association for Computational Linguistics.

Phil Blunsom, Trevor Cohn, Chris Dyer, and Miles Osborne. 2009a. A Gibbs sampler for phrasal synchronous grammar induction. In *ACL2009*, Singapore, August.

Phil Blunsom, Trevor Cohn, and Miles Osborne. 2009b. Bayesian synchronous grammar induction. In D. Koller, D. Schuurmans, Y. Bengio, and L. Bottou, editors, *Advances in Neural Information Processing Systems 21*, pages 161–168. MIT Press.

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

Colin Cherry and Dekang Lin. 2006. Soft syntactic constraints for word alignment through discriminative training. In *Proceedings of COLING/ACL*. Association for Computational Linguistics.

Colin Cherry and Dekany 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.

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

Trevor Cohn, Phil Blunsom, and Sharon Goldwater. 2010. Inducing tree-substitution grammars. *Journal of Machine Learning Research*, pages 3053–3096.

John DeNero and Dan Klein. 2010. Discriminative modeling of extraction sets for machine translation. In *The 48th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies (ACL)*.

Aria Haghighi, John Blitzer, and Dan Klein. 2009. Better word alignments with supervised itg models. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP*, Suntec, Singapore. Association for Computational Linguistics.

Mark Johnson, Thomas L. Griffiths, and Sharon Goldwater. 2007a. Adaptor grammars: A framework for specifying compositional nonparametric bayesian models. In B. Schölkopf, J. Platt, and T. Hoffman, editors, *Advances in Neural Information Processing Systems 19*, pages 641–648. MIT Press, Cambridge, MA.

Mark Johnson, Thomas L Griffiths, and Sharon Goldwater. 2007b. 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)*, pages 139–146.

Philipp Koehn, Franz Josef Och, and Daniel 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)*, pages 81–88, Edmonton, Canada, May.

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.

Daniel Marcu and William 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)*, pages 133–139, Philadelphia, July. Association for Computational Linguistics.

Daniel Marcu, Wei Wang, Abdessamad Echihabi, and Kevin 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)*, pages 44–52, Sydney, Australia, July.

Graham Neubig, Taro Watanabe, Eiichiro Sumita, Shinsuke Mori, and Tatsuya Kawahara. 2011. An unsupervised model for joint phrase alignment and extraction. In *The 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies (ACL-HLT)*, pages 632–641, Portland, Oregon, USA, 6.

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.

Adam Pauls, Dan Klein, David Chiang, and Kevin Knight. 2010. Unsupervised syntactic alignment with inversion transduction grammars. In *Proceedings of the North American Conference of the Association for Computational Linguistics (NAACL)*. Association for Computational Linguistics.

Slav Petrov, Aria Haghighi, and Dan Klein. 2008. Coarse-to-fine syntactic machine translation using language projections. In *Proceedings of EMNLP*. Association for Computational Linguistics.

M. T. Pilevar and H. Faili. 2011. Tep: Tehran english-persian parallel corpus. In *Proc. International Conference on Intelligent Text Processing and Computational Linguistics (CICLing)*.

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.

Y. W. Teh. 2006. A hierarchical Bayesian language model based on Pitman-Yor processes. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics*, pages 985–992.

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

Hao Zhang and Daniel Gildea. 2005. Stochastic lexicalized inversion transduction grammar for alignment. In *Proceedings of the 43rd Annual Conference of the Association for Computational Linguistics (ACL)*. Association for Computational Linguistics.

Hao Zhang, Chris Quirk, Robert C. Moore, and Daniel Gildea. 2008. 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)*, pages 97–105, Columbus, Ohio, June.