Bayesian Extraction of Minimal SCFG Rules for Hierarchical Phrase-based Translation

Baskaran Sankaran
Simon Fraser University
Burnaby BC, Canada
baskaran@cs.sfu.ca

Gholamreza Haffari
Monash University
Melbourne, Australia
reza@monash.edu

Anoop Sarkar
Simon Fraser University
Burnaby BC, Canada
anoop@cs.sfu.ca

Abstract

We present a novel approach for extracting a minimal synchronous context-free grammar (SCFG) for Hiero-style statistical machine translation using a non-parametric Bayesian framework. Our approach is designed to extract rules that are licensed by the word alignments and heuristically extracted phrase pairs. Our Bayesian model limits the number of SCFG rules extracted, by sampling from the space of all possible hierarchical rules; additionally our informed prior based on the lexical alignment probabilities biases the grammar to extract high quality rules leading to improved generalization and the automatic identification of commonly re-used rules. We show that our Bayesian model is able to extract minimal set of hierarchical phrase rules without impacting the translation quality as measured by the BLEU score.

1 Introduction

Hierarchical phrase-based (Hiero) machine translation (Chiang, 2007) has attracted significant interest within the Machine Translation community. It extends phrase-based translation by automatically inferring a synchronous grammar from an aligned bitext. The synchronous context-free grammar links non-terminals in source and target languages. Decoding in such systems employ a modified CKY-parser that is integrated with a language model.

The primary advantage of Hiero-style systems lie in their unsupervised model of syntax for translation: allowing long-distance reordering and capturing certain syntactic constructions, particularly those that involve discontiguous phrases. It has been demonstrated to be a successful framework with comparable performance with other statistical frameworks and suitable for large-scale corpora (Zollmann et al., 2008). However, one of the major difficulties in Hiero-style systems has been on learning a concise and general synchronous grammar from the bitext.

While most of the research in Hiero-style systems is focused on the improving the decoder, and in particular the link to the language model, comparatively few papers have considered the inference of the probabilistic SCFG from the word alignments. A majority of the systems employ the classic rule-extraction algorithm (Chiang, 2007) which extracts rules by replacing possible sub-spans (permitted by the word alignments) with a non-terminal and then using relative frequencies to estimate the probabilistic synchronous context-free grammar. One of the issues in building Hiero-style systems is in managing the size of the synchronous grammar. The original approach extracts a larger number of rules when compared to a phrase-based system on the same data leading to practical issues in terms of memory requirements and decoding speed.

Extremely large Hiero phrase tables may also lead to statistical issues, where the probability mass has to be shared by more rules: the probability $p(e|f)$ has to be shared by all the rules having the same source side string f, leading to fragmentation and resulting in many rules having very poor probability.

Approaches to improve the inference (the induction of the SCFG rules from the bitext) typically follows two streams. One focusses on filtering the extracted hierarchical rules either by removing redundancy (He et al., 2009) or by filtering rules based on certain patterns (Iglesias et al., 2009), while the other stream is concerned about alternative approaches for learning the synchronous grammar (Blunsom et al., 2008; Blunsom et al., 2009; de Gispert et al., 2010). This paper falls under the latter category and we use a non-parametric Bayesian approach for rule extraction for Hiero-style systems. Our objective in this paper is to provide a principled

Proceedings of the 6th Workshop on Statistical Machine Translation, pages 533–541,
Edinburgh, Scotland, UK, July 30–31, 2011.

rule extraction method using a Bayesian framework that can extract the minimal SCFG rules without reducing the BLEU score.

2 Motivation and Related Work

The large number of rules in Hiero-style systems leads to slow decoding and increased memory requirements. The heuristic rule extraction algorithm (Chiang, 2007) introduces redundant monotone composed rules (He et al., 2009) in the SCFG grammar. The research on Hiero rule extraction falls into two broad categories: i) rule reduction by eliminating a subset of rules extracted by the heuristic approach and ii) alternate approaches for rule extraction.

There have been approaches to reduce the size of Hiero phrase table, without significantly affecting the translation quality. He et. al. (2009) proposed the idea of discarding monotone composed rules from the phrase table that can instead be obtained dynamically by combining the minimal rules in the same order. They achieve up to 70% reduction in the phrase table by discarding these redundant rules, without appreciable reduction in the performance as measured by BLEU. Empirically analyzing the effectiveness of specific rule patterns, (Iglesias et al., 2009) show that some patterns having over 95% of the total SCFG rules can be safely eliminated without any reduction in the BLEU score.

Along a different track, some prior works have employed alternate rule extraction approaches using a Bayesian framework (DeNero et al., 2008; Blunsom et al., 2008; Blunsom et al., 2009). (DeNero et al., 2008) use a Maximum likelihood model of learning phrase pairs (Marcu and Wong, 2002), but use sampling to compute the expected counts of the phrase pairs for the E-step. Other recent approaches use Gibbs sampler for learning the SCFG by exploring a fixed grammar having pre-defined rule templates (Blunsom et al., 2008) or by reasoning over the space of derivations (Blunsom et al., 2009).

We differ from earlier Bayesian approaches in that our model is guided by the word alignments to reason over the space of the SCFG rules and this restricts the search space of our model. We believe the word alignments to encode information, useful for identifying the good phrase-pairs. For example, several attempts have been made to learn a phrasal translation model directly from the bitext without the word alignments (Marcu and Wong, 2002; DeNero et al., 2008; Blunsom et al., 2008), but without any clear breakthrough that can scale to larger corpora.

Our model exploits the word alignment information in the form of lexical alignment probability in order to construct an informative prior over SCFG rules and it moves away from a heuristic framework, instead using a Bayesian non-parametric model to infer a minimal, high-quality grammar from the data.

3 Model

Our model is based on similar assumptions as the original Hiero system. We assume that the bitext has been word aligned, and that we can use that word alignment to extract *phrase pairs*.

Given the word alignments and the heuristically extracted phrase pairs R_p, our goal is to extract the minimal set of *hierarchical* rules R_g that would best explain R_p. This is achieved by inferring a distribution over the derivations for each phrase pair, where the set of derivations collectively specify the grammar. In the following, we denote the sequence of derivations for the set of phrase pairs by $\mathbf{r}$, which is composed of grammar rules r. We will essentially read off our learned grammar from the sequence of derivations $\mathbf{r}$.

Our non-parametric model reasons over the space of the (hierarchical and terminal) rules and samples a set of rules by employing a prior based on the alignment probability of the words in the phrase pairs. We hypothesize that the resulting grammar will be compact and also will explain the phrase pairs better (the SCFG rules will maximize the likelihood of producing the entire set of observed phrase pairs).

Using Bayes' rule, the posterior over the derivations $\mathbf{r}$ given the phrase pairs R_p can be written as:

$$P(\mathbf{r}|R_p) \propto P(R_p|\mathbf{r})P(\mathbf{r}) \quad (1)$$

where $P(R_p|\mathbf{r})$ is equal to one when the sequence of rules $\mathbf{r}$ and phrase-pairs R_p are consistent, i.e. $\mathbf{r}$ can be partitioned into derivations to compose the set of phrase-pairs such that the derivations respect

the given word alignments; otherwise $P(R_p|\mathbf{r})$ is zero. The overall structure of the model is analogous to the Bayesian model for inducing Tree Substitution Grammars proposed by Cohn et al. (2009). Note that, our model extracts hierarchical rules for the word-aligned phrase pairs and not for the sentences.

Similar to the other Hiero-style systems, we use two types of rules: *terminal* and *hierarchical* rules. For each phrase-pair, our model either generates a terminal rule by *not* segmenting the phrase-pair, or decides to *segment* the phrase-pair and extract some rules.

Though it is possible to segment phrase-pairs by two (or more) non-overlapping spans, we propose a simpler model in this paper and restrict the hierarchical rules to contain only one non-terminal (unlike the case of classic Hiero-style grammars containing two non-terminals). This simpler model, samples the space of derivations and identifies a sub-span for introducing the non-terminal, which can be expressed as *terminal rules* (it is *not* decomposed further). Figure 1 shows an example phrase-pair with the Viterbi-best word alignment and Figure 2 shows two possible derivations for the same phrase-pair with the non-terminals introduced at different subspans. It can be seen that the sub-phrase corresponding to the non-terminal span X_1 is directly written as a terminal rule and is not decomposed further.

While the resulting model is slightly weaker than the original Hiero grammar, it should be noted our simpler model *does* allow reordering and discontiguous alignments. For example our model includes rules such as, $X \rightarrow (\alpha X_1 \beta, \alpha' \beta' X_1)$, which can capture phrases like (*not* X_1*, ne* X_1 *pas*) in the case of English-French translation. In terms of the reordering, our model lies in between the hierarchical phrase-based and phrase-based models. To summarize, the segmentation of each phrase-pair in our model results in two rules: a hierarchical rule with one nonterminal as well as a terminal rule.

More specifically, the generative process for generating a phrase pair x from the grammar rules may have two steps as follows. In the first step, the model decides on the type of the rule $t_x \in \{\text{TERMINAL}, \text{HIERARCHICAL}\}$ used to generate the phrase-pair based on a Bernoulli distribution, having a prior γ coming from a Beta distribution:

$$t_x \sim \text{Bernoulli}(\gamma)$$
$$\gamma \sim \text{Beta}(l_x, 0.5)$$

The lexical alignment probability l_x controls the tendency for extracting hierarchical rules from the phrase-pair x. For a given phrase-pair, l_x is computed by taking the (geometric or arithmetic) average of the reverse and forward alignment probabilities, which we explain later in this section. Integrating out γ gives us the conditional probabilities of choosing the rule type t_x as:

$$p(t_{term}|x) \propto n^x_{term} + l_x \quad (2)$$
$$p(t_{hier}|x) \propto n^x_{hier} + 0.5 \quad (3)$$

where n^x_{term} and n^x_{hier} denote the number of terminal or hierarchical rules, among the rules extracted so far from the phrase-pair x during the sampling.

In the second step, if the rule type $t_x = \text{HIERARCHICAL}$, the model generates the phrase-pair by sampling from the hierarchical and terminal rules. We use a Dirichlet Process (DP) to model the generation of hierarchical rules r:

$$G \sim DP(\alpha_h, P_0(r))$$
$$r \sim G$$

Integrating out the grammar G, the predictive distribution of a hierarchical rule r_x for generating the current phrase-pair (conditioned on the rules from the rest of the phrase-pairs) is:

$$p(r_x|r^{-x}, \alpha_h, P_0) \propto n^{-x}_{r_x} + \alpha_h P_0(r_x) \quad (4)$$

where $n^{-x}_{r_x}$ is the count of the rule r_x in the rest of the phrase-pairs that is represented by r^{-x}, P_0 is the base measure, and α_h is the concentration parameter controlling the model's preference towards using an existing hierarchical rule from the cache or to create a new rule sanctioned by the base distribution. We use the lexical alignment probabilities of the component rules as our base measure P_0:

$$P_0(r) = \Big[\Big(\prod_{(k,l) \in a} p(e_l|f_k) \Big)^{\frac{1}{|a|}} \Big(\prod_{(k,l) \in a} p(f_k|e_l) \Big)^{\frac{1}{|a|}} \Big]^{\frac{1}{2}} \quad (5)$$

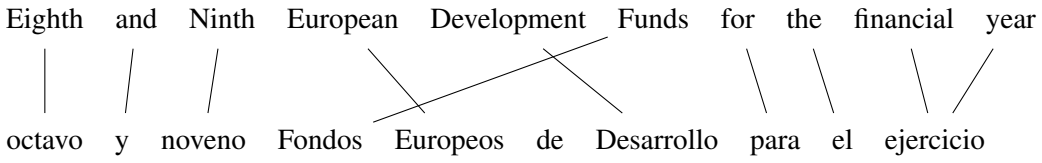

Figure 1: An example *phrase-pair* with Viterbi alignments

$X \rightarrow$ (Eighth and Ninth X_1 for the financial year, octavo y noveno X_1 para el ejercicio)
$X \rightarrow$ (*European Development Funds*, *Fondos Europeos de Desarrollo*)

$X \rightarrow$ (Eighth and Ninth X_1, octavo y noveno X_1)
$X \rightarrow$ (*European Development Funds for the financial year*,
Fondos Europeos de Desarrollo para el ejercicio)

Figure 2: Two possible derivations of the phrase-pair in Figure 1

where a is the set of alignments in the given sub-span; if the sub-span has multiple Viterbi alignments from different phrase-pairs, we consider the union of all such alignments. DeNero et al. (2008) use a similar prior- geometric mean of the forward and reverse IBM-1 alignments. However, we use the product of geometric means of the forward and reverse alignment scores. We also experimented with the arithmetic mean of the lexical alignment probabilities. The lexical prior l_x in the first step can be defined similarly. We found the particular combination of, 'arithmetic mean' for the lexical prior l_x (in the first step) and 'geometric mean' for the base distribution P_0 (in the second step) to work better, as we discuss later in Section 5.

Assuming the heuristically extracted phrase pairs to be the input to our inference algorithm, our approach samples the space of rules to find the best possible segmentation for the sentences as defined by the cache and base distribution. We explore a subset of the space of rules being considered by (Blunsom et al., 2009) — i.e., only those rules satisfying the word alignments and heuristically grown phrase alignments.

4 Inference

We train our model by using a Gibbs sampler – a Markov Chain Monte Carlo (MCMC) method for sampling one variable in the model, conditional to the other variables. The sampling procedure is repeated for what is called a long Gibbs chain spanning several iterations, while the counts are collected at fixed *thin* intervals in the chain. As is common in the MCMC procedures, we ignore samples from a fixed number of initial *burn-in* iterations, allowing the model to move away from the initial bias. The rules in the final sampler state at the end of the Gibbs chain along with their counts averaged by the number of thin iterations become our translation model.

In our model, a sample for a given phrase pair corresponds either to its terminal derivation or two rules in a hierarchical derivation. The model samples a derivation from the space of derivations that are consistent with the word alignments. In order to achieve this, we need an efficient way to enumerate the derivations for a phrase pair such that they are consistent with the alignments. We use the linear time algorithm to maximally decompose a word-aligned phrase pair, so as to encode it as a compact alignment tree (Zhang et al., 2008).

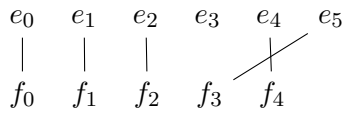

Figure 3: Example phrase pair with alignments.

For a phrase-pair with a given alignment as shown in Figure 3, Zhang et al. (2008) generalize the $\mathcal{O}(n+K)$ time algorithm for computing all K common intervals of two different permutations of length n. The contiguous blocks of the alignment are captured as the nodes in the alignment tree and the tree structure for the example phrase pair in Figure 3 is shown in Figure 4. The italicized nodes form a left-branching chain in the alignment tree and the sub-spans of this chain also lead to alignment nodes that are not explicitly captured in the tree (Please refer to Zhang et al. (2008) for details). In our work, each node in the tree (and also each sub-span in the left-branching chain) corresponds to an *aligned source-target sub-span* within the phrase-pair, and is a potential site for introducing the non-terminal X to generate hierarchical rules.

Given this alignment tree for a phrase pair, a derivation can be obtained by introducing a non-terminal at some node n_d in the tree and re-writing the span rooted at n_d as a separate rule. As mentioned earlier, we compute the derivation probability as a product of the probabilities of the component rules, which are computed using the Equation 4.

We initialize the sampler by using our lexical alignment prior and sampling from the distribution of derivations as suggested by the priors. We found this to perform better in practice, than a naive sampler without an initializer.

At each iteration, the Gibbs sampler processes the phrase pairs in random order. For each phrase pair R_p, it visits the nodes in the corresponding alignment tree and computes the posterior probability of the derivations and samples from this posterior distribution. To speedup the sampling, we store the pre-computed alignment tree for the phrase pairs and just recompute the derivation probabilities based on the sampler state at every iteration. While the sampler state is updated with the counts at each iteration, we accumulate the counts only at fixed intervals in the Gibbs chain. In applying the model for decoding, we use the grammar from the final sampler state.

Since our model includes only one hyperparameter α_h, we tune its value manually by empirically experimenting on a small set of initial phrase pairs. We keep for future work the task of automatically tuning for hyper-parameter values by sampling.

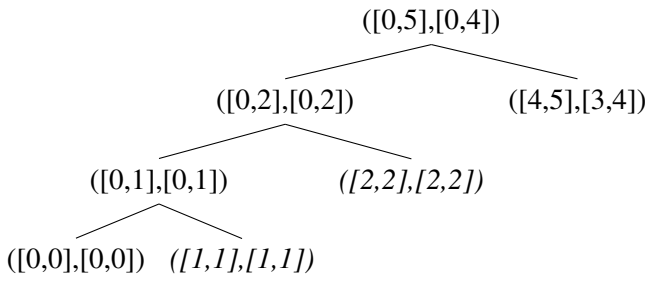

Figure 4: Decomposed alignment tree for the example alignment in Fig. 3.

5 Experiments

We use the English-Spanish data from WMT-10 shared task for the experiments to evaluate the effectiveness of our Bayesian rule extraction approach. We used the entire shared task training set except the UN data for training translation model and the language model was trained with the same set and an additional 2 million sentences from the UN data, using SRILM toolkit with Knesser-Ney discounting. We tuned the feature weights on the WMT-10 dev-set using MERT (Och, 2003) and evaluate on the test set by computing lower-cased BLEU score (Papineni et al., 2002) using the WMT-10 standard evaluation script.

We use *Kriya* – an in-house implementation of hierarchical phrase-based translation written predominantly in Python. Kriya supports the entire translation pipeline of SCFG rule extraction and decoding with cube pruning (Huang and Chiang, 2007) and LM integration (Chiang, 2007). We use the 7 features (4 translation model features, extracted rules penalty, word penalty and language model) as is typical in Hiero-style systems. For tuning the feature weights, we have adapted the MERT implementation in Moses[1] for use with Kriya as the decoder.

We started by training and evaluating the two baseline systems using i) two non-terminals and ii) one non-terminal, which were trained using the conventional heuristic extraction approach. For the baseline with one non-terminal, we modified the heuristic rule extraction algorithm appropriately[2].

[1] www.statmt.org/moses/

[2] Given an initial phrase pair, the algorithm would introduce a non-terminal for each sub-span consistent with the alignments and extract rules corresponding to each sub-span. The con-

Experiment	# of rules filtered for devset (in millions)	BLEU
Baseline (w/ 2 non-terminals)	52.36	**27.45**
Baseline (w/ 1 non-terminal)	22.09	26.71
Pattern-based filtering†	18.78	24.61
1 non-terminal; monotone & non-monotone	10.36	24.17
1 non-terminal; non-monotone	3.62	23.99

Table 1: Kriya: Baseline and Filtering experiments. †: This is the initial rule set used in Iglesias et al. (2009) obtained by greedy filtering. Rows 4 and 5 represents the filtering that uses single non-terminal rules with row 4 allowing monotone rules in addition to the non-monotone (reordering) rules.

As part of the baseline methods to be applied to minimize the number of SCFG rules, We also wanted to assess the effect of a simpler rule filtering, where the idea is to filter the heuristically extracted rules based on certain patterns. Our first baseline filtering strategy uses the heuristic methods in Iglesias et al. (2009) in order to minimize the number of rules[3]. For the other baseline filtering experiments, we retained only one non-terminal rules and then further limited it by retaining only non-monotone one non-terminal rules; in both cases the terminal rules were retained.

Table 1 shows the results for baseline and the rule filtering experiments. Restricting rule extraction to just one non-terminal doesn't affect the BLEU score significantly and this justifies the simpler model used in this paper. Secondly, we find significant reduction in the BLEU for the pattern-based filtering strategy and this is because we only use the initial rule set obtained by greedy filtering without augmenting it with other specific patterns. The other two filtering methods reduced the BLEU further but not significantly. The second column in the table gives the number of SCFG rules filtered for the devset, which is typically much less than the full set of rules. We later use this to put in perspective the effective reduction in the model size achieved by our Bayesian model. We can ideally compare our Bayesian rule extraction using Gibbs sampling with the baselines and the filtering approaches. However, running our Gibbs sampler on the full set of phrase pairs demand sampling to be distributed, possibly with approximation (**?**; **?**), which we reserve for our future work.

In this work, we focus on evaluating our Gibbs sampler on reasonable sized set of phrase pairs with corresponding baselines. We filter the initial phrase pairs based on their frequency using three different thresholds, viz. 20, 10 and 3- resulting in smaller sets of initial phrase pairs because we throw out infrequent phrase pairs (the threshold-20 case is the smallest initial set of phrase pairs). This allows us to run our sampler as a stand-alone instance for the three sets, obviating the need for distributed sampling.

Table 2 shows the number of unique phrase pairs in each set. While, the filtering reduces the number of phrase pairs to a small fraction of the total phrase pairs, it also increases the unknown words (OOV) in the test set by a factor between 1.8 and 3. In order to address this issue due to the OOV words, we additionally added *non-decomposable phrase pairs* having just one word at either source or target side,

Phrase-pairs set	# of Unique phrase-pairs	Testset OOV
All phrase-pairs	110782174	1136
Threshold-20	292336	3735
Threshold-10	606590	3056
Threshold-3	2689855	2067

Table 2: Phrase-pair statistics for different frequency threshold

straints relating to two non-terminals (such as, no adjacent non-terminals in source side) does not apply for the one non-terminal case.

[3]It should be noted that we didn't use the augmentations to the initial rule set (Iglesias et al., 2009) and our objective is to find the impact of the filtering approaches.

Experiment	Threshold-20	Threshold-10	Threshold-3
Baseline (w/ 2 non-terminals)	24.30	25.96	26.34
Baseline (w/ 1 non-terminal)	**24.00**	**25.90**	**26.83**
Bayesian rule extraction	23.39	24.30	25.22

Table 3: BLEU scores: Heuristic vs Bayesian rule extraction

Experiment	Rules Extracted (in millions)		Reduction
	Heuristic (1 nt)	Bayesian	
Threshold-20	1.93 (0.117)	1.86 (0.07)	3.57 (38.34)
Threshold-10	2.91 (1.09)	2.10 (0.28)	27.7 (73.95)
Threshold-3	7.46 (5.64)	2.45 (0.71)	**67.17 (87.28)**

Table 4: Model compression: Heuristic vs Bayesian rule extraction

Priors	α_h	BLEU
Arith + Arith means	0.5	22.46
Arith + Geom means	0.5	**23.39**
Geom + Arith means	0.5	22.96
Arith + Geom means	0.5	22.83
Arith + Geom means	0.1	22.88
Arith + Geom means	0.2	22.97
Arith + Geom means	0.3	22.98
Arith + Geom means	0.4	22.69
Arith + Geom means	0.5	**23.39**
Arith + Geom means	0.6	22.89
Arith + Geom means	0.7	22.82
Arith + Geom means	0.8	22.82
Arith + Geom means	0.9	22.67

Table 5: Effect of different priors and α_h on Threshold-20 set. The two priors correspond to the lexical prior l_x in the first step and the base distribution P_0 in the second step.

as coverage rules. The coverage rules (about 1.8 million) were added separately to the SCFG rules induced by both heuristic algorithm and Gibbs sampler. This is justified because we only add the rules that can not be decomposed further by both rule extraction approaches. However, note that both approaches can independently induce rules that overlap with the coverage rules set and in such cases we simply add the original corpus count to the counts returned by the respective rule extraction method.

The Gibbs sampler considers the phrase pairs in random order at each iteration and induces SCFG rules by sampling a derivation for each phrase pair. Given a phrase pair x with raw corpus frequency f_x, we simply scale the count for its sampled derivation $\mathbf{r}$ by its frequency f_x. Alternately, we also experimented with independently sampling for each instance of the phrase pair and found their performances to be comparable. Sampling phrase pairs once and then scaling the sampled derivation, help us to speed up the sampling process. In our experiments, we ran the Gibbs sampler for 2000 iterations with a burn-in period of 200, collecting counts every 50 iterations. We set the concentration parameter α_h to be 0.5 based on our experiments detailed later in this section.

The BLEU scores for the SCFG learned from the Gibbs sampler are shown in Table 3. We first note that, the threshold-20 set has lower baseline BLEU than threshold-10 and threshold-3 sets, as can be expected because threshold-20 set uses a much smaller subset of the full set of phrase pairs to extract hierarchical rules. The Bayesian approach results in a maximum BLEU score reduction of 1.6 for the sets using thresholds 10 and 3, compared to the one non-terminal baseline. The two non-terminal baseline is also provided to place our results in perspective.

Table 4 shows the model size, including the coverage rules for the two rule extraction approaches. The number of extracted rules, excluding the coverage rules are shown within the parenthesis. The last column shows the reduction in the model size for both with and without the coverage rules; yielding a maximum absolute reduction of 67.17% for the

threshold-3 phrase pairs set. It can be seen that the number of rules are far fewer than the rules extracted using the baseline heuristic methods for filtering detailed in Table 1. Interestingly, we obtain a smaller model size, even as we decrease the threshold to include more initial phrase pairs used as input to the inference procedure, e.g. a 67.17% reduction over the rules extracted from the threshold-3 phrase pairs v.s. a 27.7% reduction for threshold-10.

These results show that our model is capable of extracting high-value Hiero-style SCFG rules, albeit with a reduction in the BLEU score. However, our current approach offers scope for improvement in several avenues, for example we can use annealing to perturb the initial sampling iterations to encourage the Gibbs sampler to explore several derivations for each phrase pair. Though this might result in slightly large models than the current ones, we still expect substantial reduction than the original Hiero rule extraction. In future, we also plan to sample the hyperparameter α_h, instead of using a fixed value.

Table 5 shows the effect of different values of the concentration parameter α_h and the priors used in the model. The order of priors in each setting correspond to the prior used in deciding the rule-type and identifying the non-terminal span for sampling a derivation. We found the geometric mean to work better in both cases. We further found that the concentration parameter α_h value 0.5 gives the best BLEU score.

6 Conclusion and Future Work

We proposed a novel method for extracting minimal set of hierarchical rules using non-parametric Bayesian framework. We demonstrated substantial reduction in the size of extracted grammar with the best case reduction of 67.17%, as compared to the heuristic approach, albeit with a slight reduction in the BLEU scores.

We plan to extend our model to handle two non-terminals to allow for better reordering. We also plan to run our sampler on the full set of phrase pairs using distributed sampling and our preliminary results in this direction are encouraging. Finally, we would like to directly sample from the Viterbi aligned sentence pairs instead of relying on the heuristically extracted phrase pairs. This can be accomplished by using a model that is closer to the Tree Substitution Grammar induction model in (Cohn et al., 2009) but in our case the model would infer a Hiero-style SCFG from word-aligned sentence pairs.

References

Phil Blunsom, Trevor Cohn, and Miles Osborne. 2008. Bayesian synchronous grammar induction. In *Proceedings of Neural Information Processing Systems-08*.

Phil Blunsom, Trevor Cohn, Chris Dyer, and Miles Osborne. 2009. A gibbs sampler for phrasal synchronous grammar induction. In *Proceedings of Association of Computational Linguistics-09*, pages 782–790. Association for Computational Linguistics.

David Chiang. 2007. Hierarchical phrase-based translation. *Computational Linguistics*, 33.

Trevor Cohn, Sharon Goldwater, and Phil Blunsom. 2009. Inducing compact but accurate tree-substitution grammars. In *Proceedings of Human Language Technologies: North American Chapter of the Association for Computational Linguistics-09*, pages 548–556. Association for Computational Linguistics.

Adrià de Gispert, Juan Pino, and William Byrne. 2010. Hierarchical phrase-based translation grammars extracted from alignment posterior probabilities. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, pages 545–554. Association for Computational Linguistics.

John DeNero, Alexandre Bouchard-Cote, and Klein Dan. 2008. Sampling alignment structure under a bayesian translation model. In *In Proceedings of Empirical Methods in Natural Language Processing-08*, pages 314–323. Association for Computational Linguistics.

Zhongjun He, Yao Meng, and Hao Yu. 2009. Discarding monotone composed rule for hierarchical phrase-based statistical machine translation. In *Proceedings of the 3rd International Universal Communication Symposium*, pages 25–29. ACM.

Liang Huang and David Chiang. 2007. Forest rescoring: Faster decoding with integrated language models. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, pages 144–151. Association for Computational Linguistics.

Gonzalo Iglesias, Adrià de Gispert, Eduardo R. Banga, and William Byrne. 2009. Rule filtering by pattern for efficient hierarchical translation. In *Proceedings of the 12th Conference of the European Chapter of the ACL (EACL 2009)*, pages 380–388. Association for Computational Linguistics.

Daniel Marcu and William Wong. 2002. A phrase-based, joint probability model for statistical machine translation. In *In Proceedings of Empirical Methods in Natural Language Processing-02*, pages 133–139. Association for Computational Linguistics.

Franz Josef Och. 2003. Minimum error rate training in statistical machine translation. In *Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics*, pages 160–167. Association for Computational Linguistics.

Kishore Papineni, Salim Roukos, Todd Ward, and Wie-Jing Zhu. 2002. BLEU: a method for automatic evaluation of machine translation. In *In Proceedings of Association of Computational Linguistics*, pages 311–318. Association for Computational Linguistics.

Hao Zhang, Daniel Gildea, and David Chiang. 2008. Extracting synchronous grammar rules from word-level alignments in linear time. In *In Proceedings of the 22nd International Conference on Computational Linguistics (COLING) - Volume 1*, pages 1081–1088. Association for Computational Linguistics.

Andreas Zollmann, Ashish Venugopal, Franz Och, and Jay Ponte. 2008. A systematic comparison of phrase-based, hierarchical and syntax-augmented statistical mt. In *Proceedings of the 22nd International Conference on Computational Linguistics (COLING) - Volume 1*, pages 1145–1152. Association for Computational Linguistics.