# Compact Rule Extraction for Hierarchical Phrase-based Translation

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

**Gholamreza Haffari**
Monash University
Clayton VIC. Australia
reza@monash.edu

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

## Abstract

This paper introduces two novel approaches for extracting compact grammars for hierarchical phrase-based translation. The first is a combinatorial optimization approach and the second is a Bayesian model over Hiero grammars using Variational Bayes for inference. In contrast to the conventional Hiero (Chiang, 2007) rule extraction algorithm , our methods extract compact models reducing model size by 17.8% to 57.6% without impacting translation quality across several language pairs. The Bayesian model is particularly effective for resource-poor languages with evidence from Korean-English translation.our knowledge, this is the first alternative to Hiero-style rule extraction that finds a more compact synchronous grammar without hurting translation performance.

## 1 Introduction

Hierarchical phrase-based statistical machine translation (Chiang, 2007) has been shown to perform competitively with phrase-based and syntax-based models in several language pairs. A major issue with hierarchical phrase-based translation has been the size of the trained translation model, which is typically several times larger than the phrase-based counterpart trained from the same dataset. This leads to over-generation, search errors and a slower decoder (de Gispert et al., 2010).

In this paper we propose two alternative approaches to induce compact Hiero grammars. Similar to the original Hiero rule extraction (Chiang, 2007), we consider the phrase pairs that are consistent with the word alignments (Och and Ney, 2004) as the starting point in this work. Our first approach learns a minimal grammar by solving a combinatorial optimization problem over a tripartite graph consisting of three types of nodes: phrase pairs, derivations, and translation rules. This is reduced to a *minimum set cover* problem and we devise a greedy approach to extract a minimal set of translation rules to cover all the phrase pairs. Our second approach, which learns a compact but not necessarily a minimal grammar, is based on a Bayesian model for generating phrase pairs from the Hiero grammar. We use Variational Bayes (VB) for inference. The Bayesian model induces a compact Hiero grammar that has comparable performance to the original Hiero grammar in terms of the translation quality, and even improves on the full Hiero grammar when faced with a small amount of bilingual training data. On different datasets, the VB method achieves a significant reduction in the grammar size. We analyze the different extracted grammars and explain why the Bayesian model works better.

## 2 Motivation

Hierarchical phrase-based translation (Chiang, 2007) model uses a particular type of synchronous context-free grammar (SCFG) over the source and target languages. Unlike typical SCFGs, the rules are lexicalized on the right hand side with at least one aligned word pair in source and target and the grammar has one non-terminal $X$. In this paper, we refer to this type of SCFG as a *Hiero grammar*.

Rule extraction in Hiero starts from the set of *initial* bilingual phrases which are extracted from the word-aligned sentence pairs and used in typical phrase-based systems (Och and Ney, 2004). The phrase extraction criterion ensures that no word in the source/target phrase is aligned to a word that is outside the target/source phrase, while enforcing at least one alignment in the phrase.

Starting from the set of bilingual phrases $\mathcal{P}$ extracted from the word aligned sentence pairs, new translation rules are created in Hiero (Chiang, 2007) by looking for sub-phrases within the larger phrase

pair and replacing it with the nonterminal $X$. As an example, consider the phrase pair along with its Viterbi word alignment shown in Figure 1. The following translation rules (among others) are extracted by Hiero, where the corresponding non-terminals in the source and target sides are co-indexed:

$X \rightarrow \langle$모금, raising$\rangle$

$X \rightarrow \langle$모금 과정, the process of raising$\rangle$

$X \rightarrow \langle$불법 대선자금, illegal presidential campaign funds$\rangle$

$X \rightarrow \langle$불법 대선자금 $X_1$, $X_1$ illegal presidential campaign funds$\rangle$

$X \rightarrow \langle X_1$ 모금 $X_2$, $X_2$ raising $X_1\rangle$

Hiero imposes a length restriction and some constraints to avoid spurious ambiguity and to limit the grammar size, i) number of non-terminals in a rule is restricted to two, ii) no adjacent non-terminals is allowed in the source side, iii) the rule must be lexicalized with at least one aligned source-target word pair, and iv) only phrase pairs without any unaligned word in the source and target phrase boundaries are allowed. The extracted rules are filtered to remove those violating any of these constraints.

In order to estimate the rule probabilities, each phrase pair is initially assigned a unit weight, which is then equally divided among all the rules extracted from the phrase pair. The rule counts are then aggregated across all the phrase pairs, and the reverse $P(e|f)$ and forward $P(f|e)$ rule probabilities are computed using relative frequency estimation of the aggregated rule counts.

In contrast to Hiero rule extraction which extracts all possible individual rules, our approach groups the rules extracted from a phrase pair according to the derivations they belong to. It then chooses rules from one or few derivations based on their coverage over *all* phrase pairs.

Figure 2 shows some possible derivations for the phrase-pair example from Figure 1. We distinguish two types of derivations: (i) Terminal derivation (TERM) which directly rewrites a phrase pair as a terminal rule (the first derivation in Figure 2), and (ii) Hierarchical derivation (HIER) consisting of a pair of rules (derivations #2 and #3 in Figure 2).

In the rest of the paper, we use $\mathcal{P}$, $\Phi$ and $\mathcal{G}$ to denote the set of initial bilingual phrases, derivations and grammar rules respectively.

## 3 Combinatorial Optimization Approach

In our first approach we pose the problem of learning a minimal Hiero grammar in the combinatorial optimization framework as follows. *To find the minimum subset of translation rules based on which at least one derivation can be constructed for each initial phrase pair*. This problem is closely related to the *minimum set cover problem* (Vazirani, 2004), a well known NP-hard problem.[1]

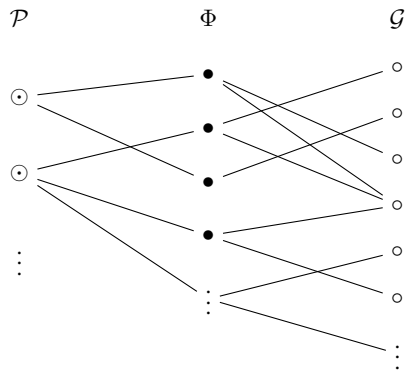



Figure 3: Tripartite graph representation of phrase-pairs ($\mathcal{P}$), derivations ($\Phi$) and grammar rules ($\mathcal{G}$)

We represent the problem as a tripartite graph $\mathcal{T}_G$ consisting of three types of vertices as in Fig 3.

• $v_x$ are vertices for phrase pairs for each phrase pair $x \in \mathcal{P}$,

• $v_{d,x}$ are vertices for derivations for each phrase pair, where $d \in \phi_x$ is a derivation from the set of all derivations $\phi_x$ for an initial phrase pair $x$,

• $v_r$ are vertices for translation rules, for each $r \in \mathcal{G}$, where $\mathcal{G}$ is the set of all constituent rules observed in the derivations of the initial phrase pairs $\mathcal{P}$.

In terms of $\mathcal{T}_G$, our aim is to select a minimal subset of rule vertices $\{v_r\}$ such that at least one derivation vertex $v_{d,x}$ is picked for each phrase-pair vertex $v_x$. We devise an efficient *greedy* algorithm to find an approximate solution for our optimization problem towards learning a compact Hiero grammar[2].

[1] Given a set of elements (called the universe) and some sets whose union comprises the universe, the *minimum set cover problem* is to identify the smallest number of sets whose union still contains all elements in the universe.

[2] In early experiments, we expressed these desiderata using an integer linear program (ILP) and its linear program (LP) re-

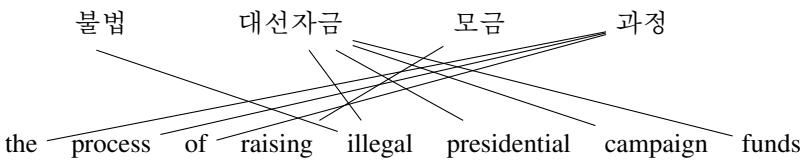


Figure 1: An example *phrase-pair* with Viterbi alignments

$X \rightarrow$ (불법 대선자금 모금 과정, the process of raising illegal presidential campaign funds)

$X \rightarrow$ (불법 대선자금 $X_1$, $X_1$ illegal presidential campaign funds)
$X \rightarrow$ *(모금 과정, the process of raising)*

$X \rightarrow$ (불법 대선자금 $X_1$ 과정, the process of $X_1$ illegal presidential campaign funds)
$X \rightarrow$ *(모금, raising)*

Figure 2: Three possible derivations (among many others) of the phrase-pair in Fig 1

The greedy method, which is listed in Algorithm 1, extracts a minimal grammar $G_m$ that explain the set of initial phrase pairs $\mathcal{P}$ by covering at least one derivation $d$ for each phrase pair $x$. We iteratively repeat the following two steps until there are no initial phrase-pair vertices in the tripartite graph $\mathcal{T}_G$: (i) Select the rule vertex which is connected to the most number of derivations in the graph, and (ii) Remove this rule and all the derivations and phrase-pair vertices reachable from this rule provided the phrase-pair vertex is covered through at least one derivation vertex. The routine degree$(v_r, \mathcal{T}_G)$ returns the in-degree of a rule vertex $v_r$ in $\mathcal{T}_G$. The function COV$(v_{d,x})$ is a Boolean function returning *true* if this vertex is not connected to any rule vertex, i.e. all the rules in derivation $d$ are present in the extracted minimal grammar $G_m$.

The greedy approach chooses one derivation for each phrase, whose component rules are added to $G_m$. For each such extracted rule, we assign it the count of the bilingual phrase from which the rule was extracted and aggregate the count across all the phrase pairs. We then simply use the relative frequency estimation for computing the conditional probabilities of the rules.

laxation. However, the size of resulting optimization problem was very large for the SMT datasets used in this paper (which are typical in size). Hence, the solution of the ILP was beyond the capacity of available off-the-shelf solvers (CPLEX).

**Algorithm 1** Greedy Algorithm for extracting Minimal Grammar

**Input:** Init phrases $\mathcal{P}$, derivations $\Phi$ and rules $\mathcal{G}$
$G_m \leftarrow \emptyset$ // minimal grammar
$C \leftarrow \mathcal{P}$ // initial phrases to be covered
**while** $C \neq \emptyset$ **do**
  $v_r \leftarrow \arg\max_{r' \in \mathcal{G}}$ degree$(v_{r'}, \mathcal{T}_G)$
  $G_m \leftarrow G_m \cup \{r\}$
  Remove $v_r$ from $\mathcal{T}_G$
  **for** $x \in C$ **do**
    **if** $\exists d \in \phi_x$ such that COV$(v_{d,x})$ is true **then**
      $C \leftarrow C - \{x\}$
      **for** $d \in \phi_x$ **do**
        Remove $v_{d,x}$ from $\mathcal{T}_G$
**Output:** Minimal grammar $G_m$

## 4 Bayesian Model for Rule Extraction

Given a set of initial phrase pairs $\mathcal{P}$ as well as a prior over the grammars, we consider Hiero grammar extraction task as the inference for the posterior over grammars in the Bayesian framework. In this section, we describe our model followed by the inference procedure using Variational Bayes.

### 4.1 Model

We represent the generation of bilingual phrases from the grammar rules as a generative process, where the process first decides the type of derivation $d$ to be either terminal ($z_d =$ TERM) or hierarchical ($z_d =$ HIER). It then identifies the constituent rules in the derivation to generate the phrase pair.

For a given phrase pair $x \in \mathcal{P}$, the probability of a derivation $d \in \phi_x$ can be expressed as:

$$P(d) \propto P(z_d) \prod_{r \in d} P(r|\mathcal{G}, \theta) \quad (1)$$

where $r$ is a rule in grammar $\mathcal{G}$, and $\theta$ are grammar parameters (the vector of rule probabilities). We assume a Dirichlet prior over the parameters:

$$\theta \sim \text{Dirichlet}(\alpha_h p_0) \quad (2)$$

where $\alpha_h$ is the concentration hyperparameter, and $p_0$ is the base measure which we construct as follows. Let $x_r = \langle f, e \rangle$ denote the phrase-pair resulted from the lexical items in the right-hand-side of a translation rule $r$. There could be many different alignments $a$ identified via learning the word alignments for different instances of $x_r$[3]. Define the forward *lf* alignment score to be (backward score *lb* is defined equivalently):

$$lf_{x_r} \propto \Big( \prod_{(m,n) \in a} p(e_n|f_m) \Big)^{\frac{1}{|a|}}$$

with $a$ being the set of alignments for the lexical items $x_r$ of rule $r$.

The base measure of a translation rule $p_0(r)$ is the arithmetic mean of the two alignment scores above. $p_0(r) \propto (lf_{x_r} + lb_{x_r})/2$.

Let $l_x$ be the geometric mean[4] of the forward and backward alignment score over an initial phrase pair $x \in \mathcal{P}$, $l_x \propto \Big(lf_x lb_x\Big)^{\frac{1}{2}}$. We place a Beta$(l_x, 0.5)$ prior over the Bernoulli distribution that decides the derivation type $z_d$ and this is normalized by the sum of lexical weights from all phrase pairs. The Beta prior prefers to consolidate a phrase pair fragment (within a larger phrase-pair) having a higher $l_x$ as a single rule. This is similar to (Sankaran et al., 2011) and we discuss the differences between the two models in Section 6.

### 4.2 Variational Inference

Variational inference (Ghahramani and Beal, 2000; Attias, 2000) is an approximation technique typically used in Bayesian settings. It is used for approximating an intractable posterior distribution $p(\Phi; \theta)$ by finding a tractable variational distribution $q(\Phi; \theta)$ over the latent variables $\Phi$ and parameters $\theta$.

Unlike the maximum likelihood (ML) or maximum a posteriori (MAP) which learn a point estimate, VB learns a *distribution* over parameters by minimizing a measure of divergence between $q$ and $p$, such as KL$(q \parallel p)$. Assuming a factorization $p(\Phi, \theta) \approx q(\Phi)q(\theta)$ enables $q(\Phi, \theta)$ to be estimated by alternately updating $q(\Phi)$ and $q(\theta)$ in an iterative setting similar to EM.

### 4.3 Variational Inference for our Model

We now describe the Variational inference procedure for our model explained earlier in Section 4.1. Using Bayes' rule, we can express the posterior over the grammar $\mathcal{G}$ given the set of bilingual phrases $\mathcal{P}$ as: $P(\mathcal{G}|\mathcal{P}) \propto P(\mathcal{G})P(\mathcal{P}|\mathcal{G})$. Specifically, we are interested in the posterior over the grammar parameters $\theta$ and the latent derivations $\Phi$ given the data and the prior. Using Variational Bayes we assume the posterior to be factorized over $\theta$ and $\Phi$ resulting in the approximate posterior as:

$$p(\theta, \Phi|\alpha_h, p_0, \mathcal{P}) \approx q(\theta|\mathbf{u})q(\Phi|\pi)$$

where $\mathbf{u}$ and $\pi$ are the parameters of the variational distributions.

The inference procedure is presented in Algorithm 2, where the parameters $\mathbf{u}^t$ and $\pi^t$ are updated iteratively. Following our assumption of Dirichlet prior over grammar parameters, we initialize $\mathbf{u}^0 := \alpha_h p_0$, which is then updated using expected rule counts in subsequent iterations. The expected rule count can be written as:

$$\mathbb{E}[r] = \sum_{d \in \phi_x} P(d|\pi^{t-1}, x) c_d(r) \quad (3)$$

where, $P(d|\pi^{t-1}, x)$ is the probability of the derivation $d$ for the phrase pair $x$ and $c_d(r)$ is the count of $r$ in derivation $d$ (the count is either 0 or 1).

The probability of a derivation in Equation 3 can be written in terms of $l_x$ and $\pi$ as:

$$P(d|\pi^{t-1}, x) \propto \begin{cases} \frac{l_x}{l_x + 0.5} \pi_r^{t-1} & \text{if } z_d = \textsc{Term} \\ \frac{0.5}{l_x + 0.5} \prod_{r' \in d} \pi_{r'}^{t-1} & \text{otherwise} \end{cases} \quad (4)$$

The probability of a derivation is normalized over all the derivations for a particular phrase pair. We fix $\alpha_h$ to be $0.5$ in our experiments, which was manually set

[3] If there are multiple alignments for $x_r$ (based on multiple initial phrase pairs), we take the union of these alignments as $a$.

[4] The reason for picking arithmetic mean for $p_0$ and geometric mean for $l_x$ is explained in (Sankaran et al., 2011).

based on a small number of trials on development data. We run Variational Bayes for fixed number of iterations (10) and read off the grammar in the last iteration together with rule pseudo counts (the expected rule counts over $e, f$ pairs). We then compute the probabilities $P(e|f)$ and $P(f|e)$ using relative frequency estimation over these pseudo counts similar to the estimation procedure in original Hiero.

**Algorithm 2** Variational Bayes Inference for learning Hiero Grammar

**Input:** Init phrases $\mathcal{P}$ and base distribution $p_0$
Get prior distribution $\mathbf{u} = \{u_r = \alpha_h p_0(r) | r \in \mathcal{G}\}$
Set $\mathbf{u}^0 = \mathbf{u}$
**for** $t = 1, 2, \ldots$ **do**
  Estimate $\pi^{t-1}$:
    $\pi_r^{t-1} \leftarrow \exp\left(\psi(u_r^{t-1}) - \psi(\sum_r u_r^{t-1})\right)$
  **for** $x \in \mathcal{P}$ **do**
    **for** $d \in \phi_x$ **do**
      Compute $P(d|\pi^{t-1}, x)$ as in (4)
  **for** $r \in \mathcal{G}$ **do**
    Compute expected rule count $\mathbb{E}[r]$ using (3)
  Estimate posteriors $u_r^t$:
    $u_r^t \leftarrow u_r^0 + \sum_{x \in R_p} \mathbb{E}[r]$
**Output:** Posterior distribution $\mathbf{u}^t$

### 4.4 VB Inference: Implementation Notes

Our model allows only one non-terminal in the rules in contrast to two non-terminals allowed by Hiero. However, we note that our model does capture reordering as well as discontiguous phrases (a key feature of Hiero). In terms of the reordering abilities, our model lies between the hierarchical phrase-based and phrase-based models.

Our model allows the unaligned source words to be attached at all possible positions in the derivation tree. This results in multiple interpretations of the unaligned words reflecting through large number of derivations, which include wider and richer rule contexts. This is analogous to the method used in (Galley et al., 2006) for context-rich syntactic translation models and we hope this to be useful in the Hiero models as well. In contrast the original Hiero grammar extraction restricts the unaligned words to be attached only to the top most position and so it can participate in just a single derivation.

To make VB inference practical, we need to efficiently enumerate all the derivations for a phrase pair such that they are consistent with the given word alignments. We use the factorization algorithm proposed by (Zhang et al., 2008) which encodes word-aligned phrase pairs as a compact alignment tree. (Zhang et al., 2008) has further details.

## 5 Experiments

**Corpora.** We use three language pairs in our experiments: Arabic-English and English-Spanish (large bilingual data conditions), and Korean-English (small bilingual data condition). Table 1 summarizes the statistics for the bilingual corpora used in this paper. For the language model, we use English Gigaword corpus (v4) for the Arabic-English and Korean-English translation tasks, and the WMT10 training data together with the UN data for the English-Spanish translation task and use 5-gram models for all language pairs.

We used the University of Rochester (Chung and Gildea, 2009) corpus for our Korean-English experiments without changing the tuning or test set splits, so our results are directly comparable to theirs. We also used the same rule-based morphological analyzer[5] as Chung and Gildea (2009) to segment the Korean side of the bitext.

**SMT Models.** We use our in-house implementation of Hiero (Chiang, 2007) with the standard features such as forward and reverse translation probabilities and lexical weights, phrase and word penalties, glue penalty and language model feature. For each experiment, we use MERT (Och, 2003) to optimize the feature weights on a tuning set, and evaluate using the corresponding optimal weights on the test set. To ensure robustness in Korean-English small data condition, we run MERT three times. The official NIST BLEU script[6] is used for computing the case-insensitive BLEU scores.

**Evaluation.** We compare our two translation grammar induction methods, based on variational Bayes (*VB*) and combinatorial optimization (*Greedy*), against the following grammars:

• *Original Hiero (2NT).* The grammar as extracted by the original rule extraction algorithm (Chiang, 2007) with two non-terminals,

• *Original Hiero (1NT).* A variant of the original

[5] http://nlp.kookmin.ac.kr/HAM/eng/main-e.html

[6] ftp://jaguar.ncsl.nist.gov/mt/resources/mteval-v13.pl

| Lang Pair | Dataset | Train/ Tune/ Test |
|---|---|---|
| *Ar-En* | ISI Ar-En corpus | 1.1 M/ 1982/ 987 |
| *En-Es* | WMT10: no UN | 1.7 M/ 5061/ 2489 |
| *Ko-En* | URochester data | 59218/ 1118/ 1118 |

Table 1: Corpus Statistics in # of sentences

rule extraction algorithm where the number of non-terminals is restricted to one[7].

We compare the model size and BLEU scores of the grammars induced by our approaches to the above two models for all the three language pairs. We also prune the VB grammar based on a count cutoff as described in Sec. 4.2 and decode using this compact grammar showing it to be competitive to the original Hiero models in terms of BLEU scores.

### 5.1 Experiments on Ar-En and En-Es

The VB inference is computationally prohibitive for Arabic-English and English-Spanish pairs due to the size of these datasets. So, we filter the set of bilingual phrases (initial phrase pairs) for these corpora based on the frequency, and run our VB inference algorithm on the filtered set of initial phrase pairs[8]. We use threshold 3 for Arabic-English, and use two thresholds (10, 20) for English-Spanish.

For Arabic-English we compare our results with heuristic rule extraction method apart from three alternative approaches for pruning Hiero grammar. First we employ pruning based on fisher significance test (Yang and Zheng, 2009) to reduce the Hiero model. We also provide results for the pattern-based filtering (Iglesias et al., 2009) that filters the grammar extracted by the original rule extraction algorithm based on certain patterns that are found to be least useful in translation or in improving the quality. And finally, we apply a fixed count cut-off on the pseudo counts of the grammar rules and eliminate all rules having pseudo counts fewer than 1.0 (we call this parameter *mincount*). This is somewhat similar to the pruning of hierarchical rules (Zollmann et al., 2008) based on a threshold, except that here we prune both lexical and hierarchical rules. Table 2 shows the BLEU scores and grammar sizes for Arabic-English.

| Grammar | BLEU | Model Size | Speed *(sent/min)* |
|---|---|---|---|
| Original Hiero (2NT) | 33.11 | 4.82 | 3.62 |
| - Yang and Zheng (2009) | 32.84 | 4.70 | 3.73 |
| - Iglesias et al. (2009) filtered | 32.52 | 3.59 | 4.99 |
| - Pruned (mincount 1.0) | 31.68 | 2.24 | 5.57 |
| Original Hiero (1NT) | 33.08 | 3.71 | 4.43 |
| - Yang and Zheng (2009) | 32.80 | 3.59 | 4.87 |
| - Iglesias et al. (2009) filtered | 32.40 | 3.43 | 5.36 |
| - Pruned (mincount 1.0) | 31.64 | 2.28 | 5.70 |
| Greedy Approach | 31.20 | 1.88 | 6.53 |
| Variational Bayes | 33.13 | 3.75 | 4.62 |
| - Pruned (mincount 1.0) | **33.05** | **2.90** | **4.87** |
| - Pruned (mincount 1.5) | 32.44 | 1.84 | 5.33 |

Table 2: Arabic-English (Threshold-3): Results. Model sizes is in millions. **Boldface** indicate the best setting of high BLEU, model size and decoding speed.

We first note that the 1NT grammar achieves comparable performance to that of the 2NT grammars and we also observe this for the other two language pairs in this paper. This shows that 1NT models does not reduce the expressive power or reordering ability and hence our Bayesian model is not handicapped by using 1NT. It also reduces the model size by 23% compared to the 2NT model.

Fisher significance pruning results in a slight drop of about 0.3 BLEU points. However it does not reduce the grammar size beyond the marginal 3.2% and this is because least frequest initial phrase-pairs are not considered in this thresholded setting. We also apply significance pruning for Korean-English- where we consider all phrase-pairs without any thresholding, and as we show later 5.2 it leads to substantial savings in the model sizes but with reduced BLEU scores. Pattern-based filtering reduces the model size by 8-26% compared to their respective baseline Hiero grammars; however the BLEU score drops by 0.7 suggesting that the blanket filtering, based purely on patterns, might actually be harmful. Using a count cutoff also significantly reduces the size of the grammars, but incurs a 1.5 point drop in the BLEU scores. The Greedy approach for combinatorial optimization worsens the BLEU score further but has a smaller model size compared to the filtering and count cutoff methods.

However, the trade-off relationship changes with our VB approach as we note that both the full VB

[7]The Hiero rule extraction algorithm can be trivially modified to limit to 1 NT grammar by replacing only one sub-phrase pair (in a larger phrase pair) with non-terminal $X$. Other rule extraction constraints are still applied.

[8]Following Sankaran et al. (2011), we add the *coverage* phrase pairs (those with non-decomposable source-target alignments) without the threshold limit to avoid OOVs (in training).

| Grammar | Threshold-10 | | Threshold-20 | |
|---|---|---|---|---|
| | BLEU | Model Size | BLEU | Model Size |
| Original Hiero (2NT) | 26.72 | 3.14 | 24.95 | 1.95 |
| Original Hiero (1NT) | 26.63 | 2.91 | 24.94 | 1.93 |
| Greedy Approach | 25.51 | 1.95 | 23.88 | 1.9 |
| Variational Bayes | 26.58 | 2.91 | *25.65* | 2.31 |
| - Pruned (mincount 1.0) | **26.55** | **2.58** | ***25.86*** | **1.97** |

Table 3: English-Spanish: Results. Model sizes is in millions. **Boldface** indicate the best setting of high BLEU and model size.

grammar and the one pruned with mincount 1.0[9] perform to the same level as the original Hiero models (without pruning). The full VB grammar has slightly larger model size than its equivalent original Hiero (1NT) model and this is due to the additional rules generated from the unaligned source words, which are attached to all possible positions in the derivation tree as we mentioned earlier. The pruned VB grammar substantially reduces the size of the grammars with the effective saving of $40.8\%$ compared to the original Hiero 2NT model. The model size can be reduced further by pruning the VB grammar with a slightly larger mincount of $1.5$ and this reduces the BLEU score modestly. This is due to the fact that VB inference produces a sharp approximation to the posterior distribution, so most of the expected counts (pseudo counts) fall below the threshold when it is slightly increased. VB provides a better trade-off between the translation quality and the model size compared to all the competing approaches. Finally we also note that the compact grammar results in 10-35% faster decoding (*speed* column in Table 2) for the pruned VB grammar compared to the original Hiero models.

We see a similar trend for the threshold-10 setting in English-Spanish experiment as seen in Table 3. Both full and pruned VB grammars achieve same translation performance as 2NT Hiero but with $17.8\%$ reduction in the grammar size. The threshold-20 setting for En-Es offers an interesting insight about the superiority of parameter estimation by VB. The pruned VB model (mincount 1) improves over the two full Hiero models by over $0.9$ BLEU points, although it uses a marginally large grammar. The *italicized* BLEU scores indicate statistically significant improvement over both 1NT and 2NT Hiero grammars, computed using bootstrap resampling with $\alpha = 0.05$. We hypothesize this to be due to improved parameter estimation using VB (see Section 5.3). Finally, the pruned VB grammar results in faster decoding compared to the Hiero 2NT models by 30% and 8% in both cases (numbers omitted due to lack of space).

[9]We explored a range of mincount values (1, 1.5 and 2) on tuning-set and present the test-set numbers that are interesting.

### 5.2 Experiments on Korean-English

Table 4 shows the BLEU scores and the grammar sizes for the different rule extraction approaches and we report the testset BLEU for the MERT run achieving the best BLEU in the tuning set. We note that the BLEU score for the Hiero 2NT and VB models are higher than the $7.27$ score obtained by Chung and Gildea (2009).

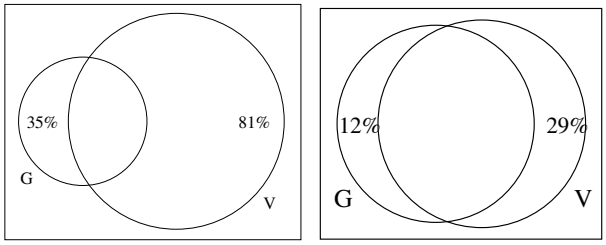
Interestingly, the greedy approach performs relatively better in this setting even though the BLEU scores of the other models are statistically significant than greedy approach. As we noted earlier significance pruning reduces the grammar by 70%, but it also hurts the translation performance as seen from the BLEU scores. This is also consistent with the more recent work by Zens et al. (2012) comparing different phrase-table pruning techniques applied to phrase-based models. Significance-based pruning were shown to perform poor compared to entropy-based pruning, even though they were better than probability-based pruning.

Unlike other languages, the Hiero 1NT and VB grammar gets a BLEU of 7.25 that is noticeably below the score of the 2NT grammar. However pruned VB grammars achieve higher BLEU scores possibly by reducing the over-generation and search error. The VB grammar pruned with threshold mincount[10] 0.25 is $57.6\%$ slimmer than the 2NT Hiero grammar. While the BLEU score of mincount 0.1 is closely behind the 0.25 setting (as also in the tuning-set), it only reduces the model size by $20.1\%$. Finally we also note that the BLEU score of mincount 0.25 is statistically significant ($\alpha = 0.1$) than 1NT Hiero.

[10]For Ko-En, we experimented with different mincount values 1.0, 0.5, 0.25 and 0.1 on the tuning-set and chose the setting (0.25) that got the highest tuning-set BLEU.

| Grammar | BLEU | Model Size | Speed *(sent/min)* |
|---|---|---|---|
| Original Hiero (2NT) | 7.53 | 2.64 | 3.82 |
| - Yang and Zheng (2009) | 6.85 | 0.75 | 5.89 |
| Original Hiero (1NT) | 7.25 | 1.83 | 4.85 |
| - Yang and Zheng (2009) | 6.93 | 0.56 | 5.97 |
| Greedy Approach | 7.04 | 1.27 | 5.25 |
| Variational Bayes | 7.28 | 2.30 | 4.73 |
| - Pruned (mincount 0.1) | 7.40 | 2.11 | 4.83 |
| - Pruned (mincount 0.25) | ***7.51*** | **1.12** | **5.41** |

Table 4: Korean-English: Results. Model sizes is in millions. **Boldface** indicate the best setting of high BLEU, model size and decoding speed.




(a) Hierarchical (b) Terminal

Figure 4: Venn diagrams of hierarchical and terminal rules in Greedy (G) and VB-pruned (V) grammars for Arabic-English (rows 4 and 9 in Table 2). The numbers indicate the % of unique rules.

### 5.3 Analysis

In this section, we investigate i) the reason for poor performance of the greedy approach and ii) why our VB inference performs to the same level as the Hiero rule extraction algorithm even after pruning. While this analysis was performed for the Ar-En, we find similar trend to hold for En-Es and Ko-En as well.

We first analyze the differences in the grammars in terms of the terminal and hierarchical rules and particularly look at the grammars generated by the VB-pruned (mincount 1.0) and that of greedy algorithm. The Venn diagrams in Figure 4 plots the overlap in the two rule types in either grammars. While $65\%$ of hierarchical rules in greedy grammar (G) are also found in the pruned VB grammar (V), only $19\%$ of the hierarchical rules in the VB grammar are included by the greedy approach. It suggests that the greedy grammar is missing crucial hierarchical rules compared to the VB grammar severely limiting its ability, for instance in reordering the phrases during decoding. Though the greedy grammar also misses $29\%$ of terminal rules found in VB grammar, its impact is minimal and as we notice in the N-best list, it uses smaller terminal rules and composing them with glue rules (this is also because greedy approach typically prefers shorter terminal rules over longer ones). We found identical trend between greedy (G) and Hiero 1NT (H) grammars as well and this suggests the poor performance of the greedy approach to be mainly due to poor *model selection*.

We also analyze the $\%$ of shared terminal and hierarchical rules for the Hiero 1NT (H) and VB-pruned (V) grammars but they had high overlap (more than $90\%$). This clearly shows that the better performance of VB grammars is not only due to its ability in model selection, but also in better parameter estimation compared to the original Hiero rule extraction algorithm. Particularly, the VB is learning a sharper distribution by moving probability mass from poor translations towards rules capturing high quality translations. Further, the high overlap (not shown due to space limitation) of the VB-pruned grammar with the Hiero 1NT grammar, indicate that the additional rules resulting from the multiple interpretations of the unaligned source words are not particularly helpful for Hiero models, unlike in syntactic models (Galley et al., 2006).

As noted earlier the Hiero rule extraction uniformly distributes the weight to the rules extracted from an initial phrase pair. These locally distributed weights are aggregated globally for each rule as these rules can be extracted from other phrase pairs as well. Therefore, it does not allow subsequent reweighting of the rules based on the global frequency of rules across the entire set of phrase pairs. In contrast, our VB inference naturally allows the rule pseudo counts to be updated at each iteration based on their global usage, thus pushing probability mass from low quality rules to high quality ones.

In order to study this quantitatively, we analyze the entropy of the source phrases (in terminal rules) that are found in both Hiero 1NT and VB grammars. For better control in the experiment, we restrict ourselves to bigram source phrases and group them into bins based on their frequency in the initial phrase pairs. We consider frequencies in 13 different intervals spaced at frequencies 2, 5, 10, 25, 50, 125, 250, 500, 1K, 2.5K, 5K, 10K, 20K and *over*. For example the two initial intervals include source phrases having frequencies $[2, 5)$ and $[5, 10)$ respectively.

For each interval, we compute the entropy of rules for unique source phrases using the conditional

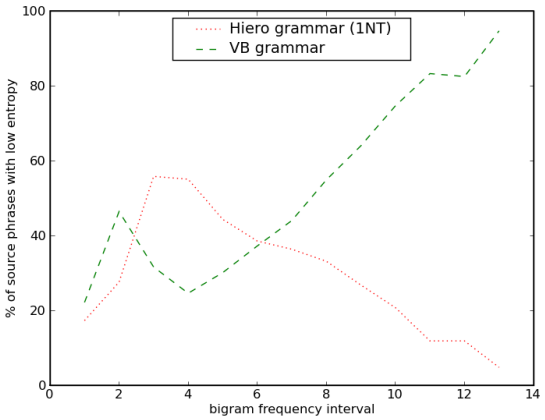



Figure 5: Entropy of the bigram source phrases of different frequencies in intervals 1-13. Intervals 1 and 2 correspond to $[2, 5)$ and $[5, 10)$ (see text for details).

probability $P(e|f)$. We compute the entropy for the source phrases that are found in both Hiero 1NT and VB grammars and aggregate this across all the source phrases within an interval. We compute the $\%$ of source phrases in VB grammar having lower entropy compared to the Hiero 1NT grammar and vice versa. Figure 5 plots the $\%$ of source phrases having lower entropy for Hiero (1NT) and VB grammars at different intervals. For most of the intervals a large percent of source phrases in VB grammar has low entropy compared to the Hiero grammar, while for 3 intervals the percentage of source phrases in Hiero grammar exceed that of VB grammar. This clearly shows that VB inference produces a sharp distribution across different frequency ranges, for both frequent and rare phrases in the training data. We also observe similar trend for trigram source phrases (skipped due to lack of space).

Next, we particularly examine the ranking of the translation options for most frequent source phrases in both grammars. We consider 100 most frequent source side $n$-grams ($n = \{1, 2, 3\}$) in the training data and compare the ranking of the translation options preferred by the two grammars (We again use $P(e|f)$ as earlier). Comparing the highest ranking translation option for these source phrases, we find both grammars to agree on the same target translation for $88.5\%$ of the $n$-grams. Additionally, in over $73\%$ of the source phrases agreeing on the same target translation, rules of the VB grammar had higher probability than the corresponding Hiero 1NT rules.

## 6 Related Works

Some earlier works have focussed on reducing the Hiero grammar size by eliminating rule redundancies in some form such as by discarding rules that can be obtained by monotonically composing the smaller rules (He et al., 2009) or by filtering the grammar, based on certain patterns of hierarchical rules in which the useful patterns were identified in a greedy fashion (Iglesias et al., 2009). Yang and Zheng (2009) applied the Fisher's exact significance test for pruning the translation model, which has been earlier used for phrase-based models (Johnson et al., 2007). As we showed in our Arabic-English experiments, our Bayesian model performs better than simple filtering approaches both in terms of BLEU and model size.

Alternately, some of the recent works have employed Bayesian techniques for inducing SCFG. Blunsom et al. (2008) proposed a generative model for deriving a sentence pair through a series of terminal and ITG-style non-terminal rules and used Variational Bayes for learning the SCFG rules. Their goal of learning a SCFG is at variance with our objective of extracting a *compact* Hiero grammar. A non-parametric Bayesian model using a Gibbs sampler to reason over the space of derivations has also been proposed (Blunsom et al., 2009). Though the model specifically uses priors to bias the grammar to be small, they do not compare the resulting grammar size. Additionally, the model suffered from weaker reordering ability and involve an additional step of extracting the SCFG rules using Hiero rule extraction algorithm on the sampled hierarchical alignments. However both these approaches use small datasets that range between 33K-300K sentence pairs. In contrast, our experiments use large datasets having 1.1M and 1.7M sentence pairs respectively for Ar-En and En-Es, with 2.2M-2.7M thresholded phrase pairs.

More recently Sankaran et al. (2011) proposed a Bayesian model for generating initial phrases and used Gibbs sampling to reason over a subset of grammar that is consistent with the heuristic phrasal alignments. While there are some similarities, our VB work is different from (Sankaran et al., 2011) in that (i) we work with a finite dimensional grammars in our model as opposed to infinite dimensional

grammars that they use, and (ii) we employ VB for inference as opposed to Gibbs sampling that they use. Further, our VB approach achieves competitive performance compared to the original Hiero rule extraction unlike the earlier work. We also present a new combinatorial optimization formulation for the induction of minimal Hiero grammars.

Variational Bayes has been used earlier for inducing probabilistic context-free grammars (PCFG) (Kurihara and Sato, 2006). Unlike theirs, we use VB for learning a Hiero-style grammar. Additionally, we do not use the free energy criterion for *model selection* as done in their work. Instead we use an informative prior for $q(\theta)$, which together with an appropriate concentration parameter $\alpha_h$, pushes the grammar towards sparsity.

## 7 Conclusion

We presented two approaches for extracting compact grammars for Hiero translation models. We demonstrated our Bayesian model using Variational inference to be competitive to the Hiero rule extraction algorithm in translation performance. It leads to an effective reduction in the model sizes, ranging $17.8 - 57.6\%$ across several language pairs. Our Bayesian model also achieves statistically significant BLEU score improvement for resource poor and small data settings. As future research, we plan to apply our Variational Bayes approach for large-scale SMT datasets as well as to extend our model to allow 2 non-terminal rules as in Hiero.

## References


Hagai Attias. 2000. A variational bayesian framework for graphical models. In *Proceedings of the NIPS*.

Phil Blunsom, Trevor Cohn, and Miles Osborne. 2008. Bayesian synchronous grammar induction. In *Proceedings of the NIPS*.

Phil Blunsom, Trevor Cohn, Chris Dyer, and Miles Osborne. 2009. A gibbs sampler for phrasal synchronous grammar induction. In *Proceedings of the ACL*.

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

Tagyoung Chung and Daniel Gildea. 2009. Unsupervised tokenization for machine translation. In *Proceedings of the EMNLP*.

Adrià de Gispert, Gonzalo Iglesias, Graeme Blackwood, Eduardo R. Banga, and William Byrne. 2010. Hierarchical phrase-based translation with weighted finite-state transducers and shallow-n grammars. *Computational Linguistics*, 36.

Michel Galley, Jonathan Graehl, Kevin Knight, Daniel Marcu, Steve DeNeefe, Wei Wang, and Ignacio Thayer. 2006. Scalable inference and training of context-rich syntactic translation models. In *Proceedings of the COLING*.

Zoubin Ghahramani and Matthew J. Beal. 2000. Variational inference for bayesian mixtures of factor analysers. In *Advances in Neural Information Processing Systems*.

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*.

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 EACL*.

Howard Johnson, Joel Martin, George Foster, and Roland Kuhn. 2007. Improving translation quality by discarding most of the phrasetable. In *Proceedings of the EMNLP-CoNLL*.

Kenichi Kurihara and Taisuke Sato. 2006. Variational bayesian grammar induction for natural language. In *International Colloquium on Grammatical Inference*.

Franz Josef Och and Hermann Ney. 2004. The alignment template approach to statistical machine translation. *Computational Linguistics*, 30:417–449.

Franz Josef Och. 2003. Minimum error rate training in statistical machine translation. In *Proceedings of the ACL*.

Baskaran Sankaran, Gholamreza Haffari, and Anoop Sarkar. 2011. Bayesian extraction of minimal scfg rules for hierarchical phrase-based translation. In *Proceedings of the Sixth Workshop on SMT*.

Vijay Vazirani. 2004. *Approximation Algorithms*. Springer.

Mei Yang and Jing Zheng. 2009. Toward smaller, faster, and better hierarchical phrase-based smt. In *Proceedings of the ACL-IJCNLP*.

Richard Zens, Daisy Stanton, and Peng Xu. 2012. A systematic comparison of phrase table pruning techniques. In *Proceedings of the EMNLP*.

Hao Zhang, Daniel Gildea, and David Chiang. 2008. Extracting synchronous grammar rules from word-level alignments in linear time. In *Proceedings of the COLING*.

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 COLING*.