Journal of Information Processing Vol.25 912–923 (Oct. 2017)

[DOI: 10.2197/ipsjjip.25.912]

Regular Paper

Hierarchical Back-off Modeling of Hiero Grammar based on Non-parametric Bayesian Model

Hidetaka Kamigaito[1,a)] Taro Watanabe[2,b)] Hiroya Takamura[3,c)] Manabu Okumura[3,d)]
Eiichiro Sumita[4,e)]

Received: October 24, 2016, Accepted: July 4, 2017

Abstract: In hierarchical phrase-based machine translation, a rule table is automatically learned by heuristically extracting synchronous rules from a parallel corpus. As a result, spuriously many rules are extracted which may be composed of various incorrect rules. The larger rule table incurs more disk and memory resources, and sometimes results in lower translation quality. To resolve the problems, we propose a hierarchical back-off model for Hiero grammar, an instance of a synchronous context free grammar (SCFG), on the basis of the hierarchical Pitman-Yor process. The model can generate compact rules and phrase pairs without resorting to any heuristics, because longer rules and phrase pairs are automatically backing off to smaller phrases under SCFG. Inference is efficiently carried out using two-step synchronous parsing of Xiao et al. combined with slice sampling. In our experiments, the proposed model achieved a higher or at least comparable translation quality against a previous Bayesian model on various language pairs: German/French/Spanish/Japanese-English. When compared against heuristic models, our model achieved comparable translation quality on a full size German-English language pair in Europarl v7 corpus with a significantly smaller grammar size; less than 10% of that for heuristic models.

Keywords: statistical machine translation, hierarchical phrase-based SMT, phrase alignments, synchronous context free grammar, Hiero grammar, non-parametric Bayesian statistics, unsupervised grammar induction

1. Introduction

Hierarchical phrase-based statistical machine translation (HPBSMT) [5] is a popular alternative to phrase-based SMT (PBSMT), in which a synchronous context free grammar (SCFG) is used as the basis of the machine translation model. With HPBSMT, a restricted form of an SCFG, i.e., Hiero grammar, is usually used and is especially suited for linguistically divergent language pairs, such as Japanese and English. In general, Hiero grammar is extracted from heuristically symmetrized word alignments [29]. This heuristic method can extract Hiero grammars without any tree structures, and is faster than other tree based models, i.e., Bayesian SCFG methods [2]. However, a rule table, i.e., a synchronous grammar, may be spuriously composed of many rules with potential errors especially when it was automatically acquired from a parallel corpus through the heuristic extraction method [5]. As a result, the increase in the rule table incurs more disk and memory resources, and sometimes results in lower translation quality. Especially, in low resource circumstances, such as a mobile translation system, a large size rule set is more problematic.

Pruning a rule table either on the basis of a significance test [16] or entropy [23], [38] used in PBSMT can be easily applied for HPBSMT. However, these methods still rely on a heuristically determined threshold parameter. Bayesian SCFG methods [2] solve the spurious rule extraction problem by directly inducing a compact rule table from a parallel corpus on the basis of a non-parametric Bayesian model without any heuristics. Training for Bayesian SCFG models infers a derivation tree for each training instance, which demands the time complexity of $O(|\mathbf{f}|^3|\mathbf{e}|^3)$ when we use dynamic programming SCFG bi-parsing [36] to a pair of source sentence $\mathbf{f}$ and target sentence $\mathbf{e}$ whose sentence lengths are $|\mathbf{f}|$ and $|\mathbf{e}|$. Gibbs sampling without bi-parsing [22] can avoid this problem, though the induced derivation trees may strongly depend on initial derivation trees. Even though we may learn a statistically sound model on the basis of non-parametric Bayesian methods, current approaches for an SCFG cannot handle rules and phrases of various granularities. The lack of various granularities may cause the generation of short length rules and phrase pairs. This is because the shorter rules and phrase pairs are more frequent than the longer one on a training dataset. The translation on short length rules and phrase pairs sometimes decreases the translation quality, because they cannnot handle the dependencies of longer distant words. For these reasons, current Bayesian approaches for an SCFG still rely on exhaustive heuristic rule extraction from the word-alignment decided by derivation trees. As long as the Bayesian SCFG meth-

1 NTT Communication Science Laboratories, Keihanna Science City, Kyoto 619–0237, Japan
2 Google Inc., Minato, Tokyo 106–6126, Japan
3 Tokyo Institute of Technology, Yokohama, Kanagawa 226–8503, Japan
4 National Institute of Information and Communication Technology, Soraku-gun, Kyoto 619–0289, Japan
a) kamigaito.hidetaka@lab.ntt.co.jp
b) tarow@google.com
c) takamura@pi.titech.ac.jp
d) oku@pi.titech.ac.jp
e) eiichiro.sumita@nict.go.jp

ods use the exhaustive heuristic rule extraction, the increase of rule set is inevitable.

To solve the problem, we propose a model on the basis of the previous work on the non-parametric Inversion Transduction Grammar (ITG) model [27] wherein phrases of various granularities are learned in a hierarchical back-off process. We extend it by incorporating arbitrary Hiero rules when backing off to smaller spans. The back-off process helps our model to generate rules and phrase pairs, which have a longer length than the previous model, since our back-off process can express longer phrase pairs by combinations of shorter rules and phrase pairs. In addition to the handling of longer phrase pairs, our back-off model can replace the several shorter rules and phrase pairs by single longer phrase pairs. For these features, our model can ease the decrasing of translation quality on compact sizes of rules and phrase pairs. However, in contrast to these benefits, the back-off process increases the inference time. For efficient inference, we use a fast two-step bi-parsing approach [37] which basically runs in a time complexity of $O(|\mathbf{f}|^3)$. Slice sampling for an SCFG [1] is used for efficiently sampling a derivation tree from a reduced space of possible derivations.

This paper is an extension of our prior work [17] with additional experiments and more detailed analysis. In particular, we add the following contents in this paper: A comparison between the previous Bayesian model with no restricted leaf nodes [*1] and our proposed back-off model; A comparison between the significance pruning method and the proposed back-off model; Analysis of actual rule and phrase pair length for each model. The results and analysis of additional experiments support our conclusion that our model achieved higher or at least comparable BLEU scores against the previous Bayesian SCFG model on the following language pairs; German/French/Spanish-English in the News-Commentary corpus, and Japanese-English in the NTCIR10 corpus. We also observed the increasing of longer phrase pairs on our back-off models on French/Spanish-English in the News-Commentary corpus, and Japanese-English in the NTCIR10 corpus. On a full size Germany-English language pair in the Europarl v7 corpus, when compared against a heuristically extracted model through the GIZA++ pipeline with and without significance pruning method, our model achieved a comparable score, and higher scores with significantly less grammar size, respectively.

2. Related Work

Various criteria have been proposed to prune a phrase table without decreasing translation quality, e.g., Fisher's exact test [16] or relative entropy [23], [38]. Although those methods are easily applied for pruning a rule table, they heavily rely on the heuristically determined threshold parameter to trade off the translation quality and decoding speed of an MT system.

Previously, EM-algorithm based generative models were exploited for generating compact phrase and rule tables. The joint phrase alignment model [24] can directly express many-to-many word aligments without heuristic phrase extraction. DeNero et al. [12] proposed the IBM Model 3 based many-to-many alignment model. The rule arithmetic method [9] can generate SCFG rules by combining other rule pairs through an inside-outside algorithm. However, those previous attempts were restricted in that the rules and phrases were induced by a heuristic combination.

Bayesian SCFG models can induce a compact model by incorporating sophisticated non-parametric Bayesian models for an SCFG, such as a Dirichlet process [2], [7], [11] or Pitman-Yor process [22], [31]. A model is learned by sampling derivation trees in a parallel corpus and by accumulating the rules in the sampled trees into the model. Due to the $O(|\mathbf{f}|^3|\mathbf{e}|^3)$ time complexity for bi-parsing a bilingual sentence, previous studies relied on bi-parsing at the initialization step, and conducted Gibbs sampling by local operators [2], [22] or sampling on fixed word alignments [7], [31]. As a result, the inference can easily result in local optimum, wherein induced derivation trees may strongly depend on the initial trees.

Xiao et al. [37] proposed a two-step approach for bi-parsing a bilingual sentence in $O(|\mathbf{f}|^3)$ in the context of inducing SCFG rules discriminatively. However, if their approach is adopted by the Markov Chain Monte Carlo algorithm (MCMC), this approach violates the detailed balance [25] due to its heuristic k-best pruning. In this case, the violation of the detailed balance is caused by the fact that the k-best pruning sometimes generate hypothesises which cannot be selected in any iteration. If the model does not satisfy the detailed balance, convergence of the model is not guaranteed. Blunsom and Cohn [1] proposed a slice sampling for an SCFG, in the same manner as that for Infinite Hidden Markov Model (iHMM) [35], which can efficiently prune a space of possible derivations on the basis of dynamic programming. Although slice sampling can prune spans without violating the detailed balance, its time complexity of $O(|\mathbf{f}|^3|\mathbf{e}|^3)$ is still impractical for a large-scale experiment. We efficiently carried out large-scale experiments on the basis of the two-step bi-parsing of Xiao et al. [37] combined with slice sampling of Blunsom and Cohn [1].

After learning a Bayesian model, it is not directly used in a decoder since it is composed of only minimum rules without considering phrases of various granularities. As a consequence, it is a standard practice to obtain word alignment from derivation trees and to extract SCFG rules heuristically from the word-aligned data [10]. The work by Neubig et al. [27] was the first attempt to directly use the learned model on the basis of a Bayesian ITG in which phrases of many granularities were encoded in the model by employing a hierarchical back-off procedure. Our work is strongly motivated by their work, but greatly differs in that our model can incorporate many arbitrary Hiero rules, not limited to ITG-style binary branching rules.

3. Model

Our proposed Back-off model is composed of a previous Bayesian SCFG Model [22]. Before we introduce our Back-off model in Section 3.3, we introduce Hiero grammar which is a target of our proposed Back-off model in Section 3.1, and the details of previous Bayesian SCFG Models in Section 3.2.

[*1] We call this setting as Gen-Relaxed in Evaluation.

3.1 Hiero Grammar

We use Hiero grammar [5], an instance of an SCFG, which is defined as a context-free grammar for two languages. Let Σ denote a set of terminal symbols in the source language, Δ a set of terminal symbols in the target language, V a set of non-terminal symbols, S a start symbol and R a set of rewrite rules. An SCFG is denoted as a tuple of $\langle \Sigma, \Delta, V, S, R \rangle$. Each rewrite rule in R is represented as $X \rightarrow \langle \alpha / \beta \rangle$ in which α is a string of non-terminals and source side terminals $(V \cup \Sigma)^*$ and β is a string of non-terminals and target side terminals $(V \cup \Delta)^*$. An example derivation in an SCFG for the sentence pair "*nihongo wo eigo ni honyaku suru koto wa muzukasii* 。 / Japanese is difficult to translate into English ." is represented as follows:

$S \rightarrow X_1$ *eigo* X_2 *muzukasii* 。 / X_1 difficult X_2 English .
$X_1 \rightarrow X_3$ *wo* / X_3 is
$X_2 \rightarrow X_4$ *honyaku suru* X_5 *wa* / X_5 translate X_4
$X_3 \rightarrow$ *nihongo* / Japanese
$X_4 \rightarrow$ *ni* / into
$X_5 \rightarrow$ *koto* / to .

Hiero grammar has the following constraints over a general SCFG:

- Phrase pairs are not allowed to contain multiple word alignments. Therefore, only the smallest phrase pairs are kept.
- The number of terminal and non-terminal symbols in each rule for both source and target sides is limited to 5.
- Each rule may contain at most two non-terminal symbols.
- Adjacent non-terminal symbols in the source side are prohibited.
- A rule must contain at least one alignment between source and target terminal symbols.

For details, refer to Ref. [5].

3.2 Bayesian SCFG Models

3.2.1 Generative Process

Previous Bayesian SCFG Models, for instance a model proposed by Levenberg et al. [22], are based on Pitman-Yor process [32] and learn SCFG rules by sampling a derivation tree for each bilingual sentence. **Figure 1** shows an example derivation tree for our running example sentence pair under the model. The generative process is represented as follows:

$$\begin{aligned} &G_X \sim P_{rule}(d_r, \theta_r, G_{r_0}), \\ &X \rightarrow \langle \alpha / \beta \rangle \sim G_X, \end{aligned} \tag{1}$$

where G_X is a derivation tree and $P_{rule}(d_r, \theta_r, G_{r_0})$ is a Pitman-Yor process [32], which is a generalization of a Dirichlet process parametrized by a discount parameter d_r, a strength parameter θ_r and a base measure G_{r_0}. The output probability of a Pitman-Yor process obeys the power-law distribution with the discount parameter, which is very common in standard NLP tasks.

The probability that a rule r_k is drawn from a model $P_{rule}(d_r, \theta_r, G_{r_0})$ is determined by a Chinese restaurant process (CRP) which is decomposed into two probability distributions. If r_k already exists in a table, we draw r_k with probability

$$\frac{c_k - d_r \cdot |\varphi_{r_k}|}{\theta_r + n_r}, \tag{2}$$

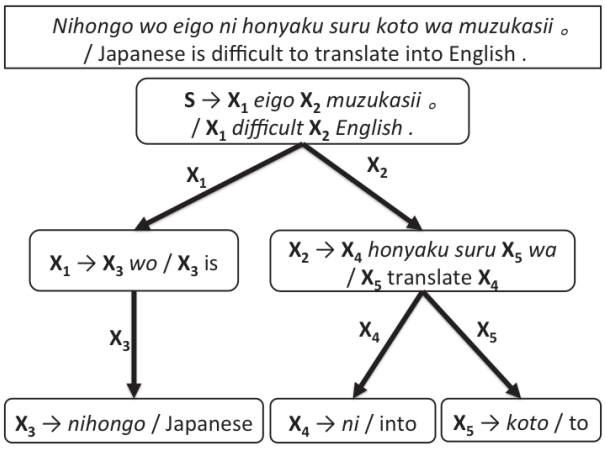


Fig. 1 Derivation tree generated from Bayesian SCFG model.

where c_k is the number of customers of r_k, n_r is the number of all customers and φ_{r_k} is the number of r_k's tables. On the other hand, if r_k is a new rule, we draw r_k with probability

$$\frac{\theta_r + d_r \cdot |\varphi_r|}{\theta_r + n_r} \cdot G_{r_0}, \tag{3}$$

where $|\varphi_r|$ is the number of tables in the model.

3.2.2 Base Measure

In the previous Bayesian SCFG model [22], the base measure for rule probability G_{r_0} is composed of four generative processes. First, the number of symbols in a source side of a rule $|\alpha|$ is generated from a Poisson distribution:

$$|\alpha| \sim Poisson(0.1). \tag{4}$$

Let $t(x)$ denote a function that returns terminals from a string x. The number of target side terminal symbols $|t(\beta)|$ is also generated from a Poisson distribution and represented as:

$$|t(\beta)| \sim Poisson(\alpha + \lambda_0), \tag{5}$$

where λ_0 is a small constant for the input distribution greater than zero. The type of symbol α_i in the source side, $type_i$, either terminal or non-terminal symbol, is determined by:

$$type_i \sim Bernoulli(\phi^{|\alpha|}), \tag{6}$$

where ϕ is a hyperparameter taking $0 < \phi < 1$. $\phi^{|\alpha|}$ is based on an intuition that shorter rules should be relatively more likely to contain terminal symbols than longer rules. Source and target terminal symbol pairs $\langle t(\alpha), t(\beta) \rangle$ are generated from the geometric means of two directional IBM Model 1 word alignment probabilities and monolingual unigram probabilities for two languages, and represented as:

$$\begin{aligned} \langle t(\alpha), t(\beta) \rangle \sim\ & (P_{uni}(t(\alpha)) P_{\overrightarrow{M1}}(t(\alpha), t(\beta)) \\ & \cdot P_{uni}(t(\beta)) P_{\overleftarrow{M1}}(t(\alpha), t(\beta)))^{\frac{1}{2}}. \end{aligned} \tag{7}$$

When $t(\alpha)$ or $t(\beta)$ is empty, we use the constant 0.01 instead of the Model 1 probabilities.

3.3 Hierarchical Back-off Model

3.3.1 Generative Process

In the previous models, the generative process is represented as

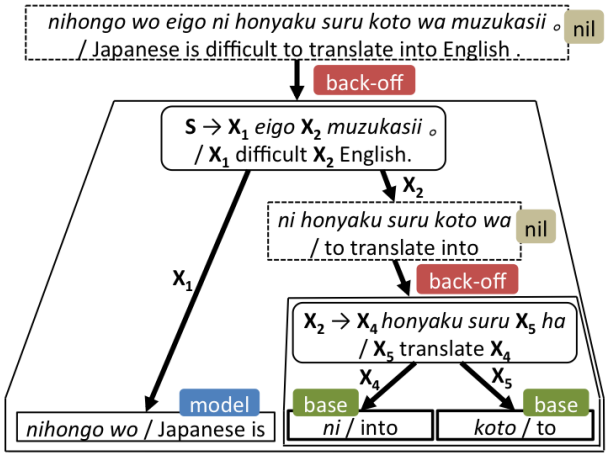


Fig. 2 Derivation tree generated from the hierarchical back-off model.

a rewrite process starting from the symbol S, which can incorporate only minimal rules. Following Neubig et al. [27], our model reverses the process by recursively backing off to smaller phrase pairs as shown in **Fig. 2**. First, our model attempts to generate a phrase pair, i.e., a sentence pair, as a derivation tree. If the model successfully generates the phrase pair, we will finish the generation process. Otherwise, a Hiero rule is generated to fallback to smaller spans represented in each non-terminal symbol X in the rule. Then, each phrase pair corresponding to each smaller span is recursively generated through our model. In Fig. 2, a phrase pair with "nil" indicates those not in our model; therefore the phrase pair is forced to back-off either by generating a new phrase pair from a base measure (base) or by falling back to smaller phrases using a Hiero rule (back-off). The recursive procedure is done until we reach phrase pairs which are generated without any back-offs. Let a discount parameter be d_p, a strength parameter be θ_p, and a base measure be G_{p_0}. More formally, the generative process is represented as follows:

$$\begin{aligned} &G_X \sim P_{rule}(d_r, \theta_r, G_{phrase}), \\ &G_{phrase} \sim P_{phrase}(d_p, \theta_p, G_X), \\ &X \to \langle \mathbf{s}/\mathbf{t} \rangle \sim G_{phrase}, \\ &X \to \langle \alpha/\beta \rangle \sim G_X, \end{aligned} \tag{8}$$

where $\mathbf{s}$ is source side terminals and $\mathbf{t}$ is target side terminals in phrase pair $\langle \mathbf{s}/\mathbf{t} \rangle$. P_{phrase} is composed of three states, i.e., model, back-off, and base, and follows a hierarchical Pitman-Yor process [34].

model:

We draw a phrase pair $\langle \mathbf{s}/\mathbf{t} \rangle$ with the probability similar to Eq. (2):

$$\frac{c_k - d_p \cdot |\varphi_{p_k}|}{\theta_p + n_p}, \tag{9}$$

where c_k is the number of customers of a phrase pair p_k and n_p is the number of all customers. Note that this state is reachable when the phrase pair $\langle \mathbf{s}/\mathbf{t} \rangle$ exists in the model in the same manner as Eq. (2).

back-off:

We will back off to smaller phrases using a rule generated by P_{rule} as follows:

$$\begin{aligned} &\frac{\theta_p + d_p \cdot |\varphi_p|}{\theta_p + n_p} \cdot \frac{c_{back} + \gamma_b \cdot G_b}{c_{back} + c_{base} + \gamma_b} \\ &\cdot P_{rule}(d_r, \theta_r, G_{phrase}) \\ &\cdot \prod_{X \in \langle \alpha/\beta \rangle} P_{phrase}(d_p, \theta_p, G_X), \end{aligned} \tag{10}$$

where c_{back} and c_{base} are the number of customers sampled from the back-off and base phrases, respectively, with a base measure G_b and hyperparameter γ_b. We use a uniform distribution for $G_b = 0.5$ since we consider only two states, back-off and base. Unlike the model state, P_{phrase} may reach this state even when a phrase pair is not in the model. The phrase pair is backed-off to smaller phrase pairs using P_{phrase} through the non-terminals in the generated rule $X \in \langle \alpha/\beta \rangle$.

base:

As an alternative to the back-off state, we may reach the base state which follows the probability distribution on the basis of the base measure G_{p_0},

$$\frac{\theta_p + d_p \cdot |\varphi_p|}{\theta_p + n_p} \cdot \frac{c_{base} + \gamma_b \cdot G_b}{c_{back} + c_{base} + \gamma_b} \cdot G_{p_0}. \tag{11}$$

In summary, $P_{phrase}(d_p, \theta_p, G_X)$ is defined as a joint probability of Eqs. (9) through (11).

3.3.2 Base Measure

The base measure for phrases G_{p_0} is composed of three generative processes, in a similar manner as Levenberg et al. [22]. The number of terminal symbols in a phrase pair in the source side, $|\mathbf{s}|$, is generated from a Poisson distribution:

$$|\mathbf{s}| \sim Poisson(0.1). \tag{12}$$

The length for the target side $|\mathbf{t}|$ is generated in the same manner as the source side of the phrase pair. The alignments between $\mathbf{s}$ and $\mathbf{t}$ are also generated in the same manner as those for the base measure in a rule.

4. Inference

For efficient inference, we propose a method to combine a fast two-step bi-parsing approach [37] and slice sampling for an SCFG [1]. Sampling for inference is conducted by a sentence-wise block sampling [1], which has a better convergence property when compared with a step-wise Gibbs sampling. We repeat the following steps given a sentence pair.

(1) Decrement the number of customers of the rules and phrase pairs used in the current derivation for the sentence pair.
(2) Bi-parse the sentence pair in a bottom up manner.
(3) Sample a new derivation tree in a top-down manner.
(4) Increment the number of customers of the rules and phrase pairs in the sampled derivation tree.

The most time-consuming step during the inference procedure is bi-parsing of a sentence pair which essentially takes $O(|\mathbf{f}|^3|\mathbf{e}|^3)$ time using a bottom up dynamic programming algorithm [36]. When a span is very large, it can easily suffer combinatorial explosion. To avoid this problem, we use a two-step slice sampling by performing the two-step bi-parsing [37] and by pruning possible derivation space [1] in each step (Algorithm 1). From lines 1 to 8, a set of word alignment is enumerated and put into a list of

Algorithm 1 Two-step slice sampling

```
1: for i ← 1, ··· , |source| do
2:   for j ← 1, ··· , |target| do
3:     cube_a ← {source_i, target_j}
4:   end for
5:   cube_a ← {source_i, null}
6:   chart ← SliceSampling(cube_a)
7:   clear cube_a
8: end for
9: for h ← 1, ··· , |source| do
10:  for all the i, j s.t j − i = h do
11:    for inferable rule, phrase from the subspans of [i, j] of all charts do
12:      cube ← rule, phrase
13:    end for
14:    chart ← SliceSampling(cube)
15:    clear cube
16:  end for
17: end for
```

word alignments $cube_a$. In addition to the arbitrary word alignment of $source_i$ to $target_j$, null word alignment is also merged into $cube_a$ (line 5). Note that word alignment considered in the algorithm is restricted to one-to-many [*2]. the set of word alignments in $cube_a$ is pruned and added to the *chart* by *SliceSampling*, where *chart* denotes a hyper-graph, representing connections of all rules and phrase pairs. In *SliceSampling*, the phrases and rules in *cube* are pruned based on a randomly sampled threshold and the remainders are added to *chart*. From lines 9 to 17, all possible phrases and rules for each span constrained by the remained word alignment are enumerated and temporally stored into a list of rules and phrase pairs *cube*. The time complexity for the word alignment enumeration from lines 1 to 8 is $O(|f||e|)$ and that for the phrase and rule enumeration from lines 9 to 17 is $O(|f|^3)$.

The key difference to the slice sampling [1] lies in lines 6 and 3 of Algorithm 1. Let $\mathbf{d}$ denote a set of derivation trees d and $\mathbf{u}$ be a set of slice variables u. In slice sampling, we prune the rules $\mathbf{r_{sp}}$ in each source span sp based on a slice variable u_{sp} corresponding to that sp. After pruning, we sample trees from the pruned space of $\mathbf{r}$. The above process is formally represented as:

$$\begin{aligned}\mathbf{u} &\sim P(\mathbf{u}|\mathbf{d}),\\ \mathbf{d} &\sim P(\mathbf{d}|\mathbf{u}),\end{aligned} \tag{13}$$

where $P(\mathbf{d}|\mathbf{u})$ is computed through sampling in a top-down manner after parsing in a bottom-up manner with Algorithm 1, and is equal to $\prod_d P(d|\mathbf{u})$. The probability $P(\mathbf{u}|\mathbf{d})$ is equal to $\prod_{sp} P(u_{sp}|\mathbf{d})$. Let r^*_{sp} denote a currently adopted rule in the span sp and $P(u_{sp}|d)$ be defined using a pruning score $Score(r^*_{sp})$ as follows:

$$Score(r_{sp_i}) = Inside(r_{sp_i}) \cdot Future(r_{sp_i}), \tag{14}$$

where $Inside(r_{sp})$ and $Future(r_{sp})$ are inside and outside probabilities for sp, respectively. Let $\mathbf{s}_{r_{sp}}$ denote a set of source side words in r_{sp}, $\mathbf{t}_{r_{sp}}$ a set of target side words in r_{sp}, $\overline{\mathbf{s}_{sp}}$ a set of words in a source sentence without $\mathbf{s}_{r_{sp}}$ and $\overline{\mathbf{t}_{sp}}$, a set of words in a target sentence without $\mathbf{t}_{r_{sp}}$. By using IBM Model 1 probabilities in two directions, $Inside(r_{sp})$ is calculated by

$$(P_{\overrightarrow{M1}}(\mathbf{s}_{sp}, \mathbf{t}_{sp}) \cdot P_{\overleftarrow{M1}}(\mathbf{s}_{sp}, \mathbf{t}_{sp}))^{\frac{1}{2}}. \tag{15}$$

We use the IBM Model 1 outside probability for future score $Future(r_{sp})$. Similarly, the future score $Future(r_{sp})$ is computed using the two directional models:

$$(P_{\overrightarrow{M1}}(\overline{\mathbf{s}_{sp}}, \overline{\mathbf{t}_{sp}}) \cdot P_{\overleftarrow{M1}}(\overline{\mathbf{s}_{sp}}.\overline{\mathbf{t}_{sp}}))^{\frac{1}{2}}. \tag{16}$$

When sp is used in the current derivation $\mathbf{d}$, slice variable u_{sp} is sampled from a uniform distribution [*3]:

$$P(u_{sp}|d) = \frac{\mathbb{I}(u_{sp} < Score(r^*_{sp}))}{Score(r^*_{sp})}, \tag{17}$$

otherwise, u_{sp} is sampled from a beta distribution if sp is not in the current derivation $\mathbf{d}$:

$$P(u_{sp}|d) = Beta(u_{sp}; a, 1.0), \tag{18}$$

where $a < 1$ is a parameter for the beta distribution. If the $Score(r_{sp_i})$ is less than u_{sp}, we prune the r_{sp_i} from *cube*. Similar to Blunsom and Cohn [1], if the span sp is not in the current derivation, the rules with a low probability are pruned according to Eq. (18). Letting r^d_{sp} denotes a rule in d with span sp, $P(d|\mathbf{u})$ is calculated by:

$$\prod_{sp \in d} \frac{P(r^d_{sp})}{\sum_{r_j \in \mathbf{r_{sp}}} P(r_j)\mathbb{I}(u_{sp} < Score(r_j))}. \tag{19}$$

In our experiments discussed in Section 6, slice sampling parameter a was set to 0.02 when calculating the future score in Eq. (16). In contrast, we used $a = 0.1$ when performing slice sampling without the future score. We empirically found that setting a lower value for a led to slower progress in learning due to a combinatorial explosion when inferencing a derivation for each sentence pair.

In the beginning of training, we do not have any derivation trees for given training data, although the derivation trees are required for estimating parameters for Bayesian models. We use the two-step parsing for generating initial derivation trees solely from base measures. K-best pruning is conducted against the score denoted by Eq. (14), which is very similar to Xiao et al. [37] [*4].

For faster bi-parsing, we run sampling in parallel in the same way as Zhao and Huang [39], in which bi-parsing is performed in parallel among the bilingual sentences in a mini-batch. The updates to the model are synchronized by incrementing and decrementing the number of customers for the bilingual sentences in the mini-batch. Note that the bi-parsing for each mini-batch is conducted on the fixed model parameters after the synchronized parameter updates.

In addition to the model parameters, hyperparameters are resampled after each training iteration following the discount and strength hyperparameter resampling in a hierarchical Pitman-Yor

[*2] In Hiero grammar, as we explained in the Section 3.1, phrase pairs are not allowed to contain multiple word alignments. This restriction may decrease the expressiveness of the previous Bayesian SCFG models. In Experiments, for fair comparisons between previous Bayesian model and our hierarchical back-off model, we removed the restriction of the multiple word alignments, and named this setting as Gen-Relaxed. In Gen-Relaxed, the expressiveness of the previous Bayesian SCFG model is the same as our hierarchical back-off model.

[*3] $\mathbb{I}(\cdot)$ is a function that returns 1 if the condition is satisfied and 0 otherwise

[*4] Note that we use $k = 30$ for k-best pruning.

process [34]. In particular, we resample $\langle d_p, \theta_p \rangle$, the pair of discount and strength parameters for phrases from a distribution:

$$\frac{[\theta_p]_{d_p}^{|\varphi_p|}}{[\theta_p]_1^{n_p}} \prod_{\langle \mathbf{s},\mathbf{t} \rangle} \prod_{k=1}^{|\varphi_p|} [1 - d_p]_1^{(c_{\langle \mathbf{s},\mathbf{t} \rangle} - 1)} \tag{20}$$

where [ ] denotes a generalized Pochhammer symbol, and $c_{\langle \mathbf{s},\mathbf{t} \rangle}$ is the number of customers of phrase pair $\langle \mathbf{s}, \mathbf{t} \rangle$. We resample the pair $\langle d_r, \theta_r \rangle$ in the same way as $\langle d_p, \theta_p \rangle$. The hyperparameter γ_b is resampled from distribution:

$$\frac{(c_{back} + \gamma_b \cdot G_b)(c_{base} + \gamma_b \cdot G_b)}{(c_{back} + c_{base} + \gamma_b)^2}, \tag{21}$$

where ϕ, used in the generative process for either terminal or non-terminal symbol $type_i \sim Bernoulli(\phi^\alpha)$, is resampled from the following distribution:

$$\prod_{\langle \alpha/\beta \rangle \in Base} Bernoulli(\phi^{|\alpha|})^{c_{\langle \alpha/\beta \rangle}}, \tag{22}$$

where $c_{\langle \alpha/\beta \rangle}$ denotes the number of customers of rule $\langle \alpha/\beta \rangle$, and $Base$ denotes a set of rules generated from the base measure. All the hyperparameters are inferred by slice sampling [26].

5. Extraction of Translation Model

In the previous work on Bayesian approaches [1], [22], it is standard practice to heuristically extract rules and phrase pairs from the word alignment derived from the derivation trees sampled from the Bayesian models. Instead of the heuristic method, we directly extract rules and phrase pairs from the learned models which are represented as Chinese restaurant tables. To limit grammar size, we include only phrase pairs that are selected at least once in the sample.

For each extracted rule or phase pair, we compute a set of feature scores used for a HPBSMT decoder; a weighted combination of multiple features is necessary in SMT since the model learned from training data may not fit well to translate an unseen test data [28]. We use the following six features; the joint model probability P_{model} is calculated by Eq. (2) for rules and by Eq. (9) for phrase pairs. The joint posterior probability $P_{posterior}(f, e)$ is estimated from the posterior probabilities for every rule and phrase pair in derivation trees through relative count estimation, motivated by Neubig et al. [27] [*5]. The joint posterior probability is considered as an approximation for those back-off scores. The conditional model probabilities in two directions, $P_{model}(f|e)$ and $P_{model}(e|f)$, are estimated by marginalizing the joint probability $P_{model}(f, e)$:

$$P_{model}(f|e) = \frac{P_{model}(f, e)}{\sum_{f'} P_{model}(f', e)}. \tag{23}$$

The inverse direction $P_{model}(e|f)$ is estimated similarly. The lexical probabilities in two directions, $P_{lex}(f|e)$ and $P_{lex}(e|f)$, are scored by IBM Model 1 probabilities between the source and target terminal symbols in rules and phrase pairs. In addition to the above features, we use Word penalty for each rule and phrase pair used in the cdec decoder [13].

[*5] Note that the correct way to decode from our model is to score every phrase pair created during decoding with back-off states, which is computationally intractable.

As indicated in previous studies [12], [21], the translation quality of generative models is lower than that of models with heuristically extracted rules and phrase pairs. DeNero et al. [12] reported that considering multiple phrase boundaries is important for improving translation quality. The generative models, in particular Bayesian models, are strict in determining phrase boundaries since their models are usually estimated from sampled derivations. As a result, translation quality is poorer when compared with a model estimated using a heuristic method. The Hiero grammar severely suffers from the phrase granularity problem and can overfit to the training data due to the flexibility of the rules.

To alleviate this problem, Neubig et al. [27] combined the derivation trees across training iterations by averaging the features for each rule and phrase pair. During the sampling process, each training iteration draws a different derivation tree for each sentence pair, and the combination of those different derivation trees can provide multiple possible phrase boundaries to the model. Inspired by the averaging over the models from different iterations, we combine them as a part of a sampling process; we treat the derivation trees acquired from different iterations as additional training data, and increment the number of the corresponding customers into our model. Hyperparameters are resampled after the merging process. The new features are directly computed from the merged model.

6. Experiments

6.1 Comparison with Previous Bayesian Model

First, we compared the previous Bayesian model [22] with our hierarchical back-off model. The details are as follows:

- **Gen** Previous Bayesian SCFG model with Hiero constraints. Due to the minimal phrase constraint of a Hiero grammar, Gen can only generate one-to-many phrase pairs. In inference steps, different to the original method, we used our proposed two step slice sampling approach without future scores.
- **Gen-Relaxed** For fair comparisons between the previous Bayesian model and our hierarchical back-off model, we removed the minimal phrase constraint of a Hiero grammar to generate many-to-many word alignments in the previous Bayesian model. Other settings are same as Gen.
- **Back** Our proposed hierarchical back-off model inferenced by the two step slice sampling approach without future scores.
- **Back+future** Our proposed hierarchical back-off model inferenced on the two step slice sampling approach with future scores.

In the German/Spanish/French-to-English translation pairs, we used WMT 2010 test set [3] for testing, WMT 2009 test set [4] for tuning, and News-Commentary-v8 corpus for training. In the Japanese-to-English translation pair, we used NTCIR10 corpus [14] for tuning, testing and training. We used the first 100 K sentence pairs of training data sets for the translation models. All sentences were tokenized, lowercased, and filtered to preserve at most 40 words on both source and target sides for training. We sampled 20 iterations for Gen and Back and combined the

Table 2 Results of translation evaluation in 100 k corpus.

		News-Commentary						NTCIR10	
		de-en		es-en		fr-en		ja-en	
Model	Sample	BLEU	SIZE	BLEU	SIZE	BLEU	SIZE	BLEU	SIZE
∗ GIZA++	-	16.66	7.07 M	23.16	6.07 M	20.79	6.25 M	26.08	3.45 M
Gen	1	15.36	397.63 k	21.10	295.69 k	19.45	311.76 k	25.73	262.45 k
	10	15.39	529.46 k	20.83	384.55 k	19.24	419.33 k	25.79	344.67 k
Gen-Relaxed	1	15.30	397.61 k	20.90	295.53 k	19.33	312.00 k	26.00	268.17 k
	10	15.31	591.62 k	21.10	433.64 k	19.43	464.61 k	25.52	378.28 k
Back	1	15.30	410.92 k	*21.43*	314.95 k	*19.74*	362.22 k	25.69	294.90 k
	10	15.42	563.80 k	*21.53*	420.15 k	19.51	497.51 k	25.63	388.87 k
Back+future	1	**15.49**	384.69 k	**21.63**	296.30 k	**19.97**	340.70 k	**25.82**	268.38 k
	10	**15.55**	579.12 k	**21.74**	429.33 k	**19.97**	513.41 k	25.41	390.23 k

Table 1 The statistics for each corpus.

		De-En	Es-En	Fr-En	Ja-En
TM	*Sentence*	100.0 k	100.0 k	100.0 k	100.0 k
TM(en)	*Word*	1.9 M	1.7 M	1.5 M	1.8 M
TM(other)	*Word*	1.9 M	1.9 M	1.8 M	2.0 M
LM(en)	*Sentence*	2.5 M	2.5 M	2.5 M	3.2 M
LM(en)	*Word*	55.6 M	55.6 M	55.6 M	27.8 M
Dev	*Sentence*	2.5 k	2.5 k	2.5 k	2.0 k
Dev(en)	*Word*	65.5 k	65.5 k	65.5 k	67.3 k
Dev(other)	*Word*	62.7 k	68.1 k	72.5 k	73.0 k
Test	*Sentence*	2.5 k	2.5 k	2.5 k	8.6 k
Test(en)	*Word*	61.9 k	61.9 k	61.9 k	310.0 k
Test(other)	*Word*	61.3 k	65.5 k	70.5 k	333.0 k

last 10 iterations for extracting the translation model [*6]. During the extraction process, we limited the source or target terminal symbol size of phrase pairs to 5. The batch size was set to 64. The language models were estimated from the all-English side of the WMT News-Commentary and europarl-v7. In NTCIR10, we simply used the all-English side of the training data. All the 5-gram language models were estimated using SRILM [33] with interpolated Kneser-Ney smoothing. The details of the corpus are presented in **Table 1** [*7]. For detailed analysis, we also evaluate Hiero grammars extracted from GIZA++ [29] grow-diag-final bidirectional alignments using Moses [18] with Hiero options. We use GIZA++ and Moses default parameters for training. Decoding was carried out using the cdec decoder [13]. Feature weights were tuned on the development data by running MIRA [6] for 20 iterations with 16 parallel. For other parameters, we used cdec's default values. The numbers reported here are the average of three tuning runs [15].

Table 2 lists the results measured using BLEU [30]. The row marked up with ∗ indicates the model using word class information [*8]. The column Sample denotes the combination size for each model. The column SIZE in the table denotes the number of the extracted grammar types composed of Hiero rules and phrase pairs. The numbers in italic denote the significance improvement from the score of 1 and 10 sample combinated Gen and Gen-Relaxed. The numbers in bold denote the score of Back+future, significantly improved from the higher score of 1 and 10 sampled combinated Back. This bold numbers shows the effectiveness of our proposed slice sampling with future score. All significance tests are performed using multeval [8] under p-value of 0.05. Back performed better than Gen and Gen-Relaxed on Spanish-English and French-English language pairs. Note that the gains were achieved with the comparable grammar size. When comparing German-English and Japanese-English language pairs, Back has no significant improvement on Gen and Gen-Relaxed. The combination of our Back with future scores during slice sampling (+future) achieved further gains over the slice sampling without future scores, and slightly decrease the grammar size, compared to Back. However, Back+future has still no significant improvement on Gen and Gen-Relaxed in German-English and Japanese-English language pairs. The sample combination has no or slight gains on the BLEU score, in spite of the increase in the grammar size. From the results, using the last one sample as a grammar is sufficient for translation quality. The performance of the Bayesian model did not match with that for the GIZA++ pipeline heuristic approach. In general, complex models, such as Gen and Back, demand larger corpus size for training, and the evaluation on such smaller corpus may not be a fair comparison, since the sampling approach can only rely on sampled derivations. Thus, we evaluate these methods on a large size corpus in the next section.

6.2 Comparison with Heuristic Extraction

As reported in Refs. [12], [21], the comparison against heuristic extraction is a challenging task. We compared the Back+future and a baseline extracted from grow-diag-final alignments (GDF) of GIZA++ using Moses with Hiero options in the German-to-English translation pair. We used GIZA++ and Moses default parameters for training. IBM Model 4 is the main model implemented in GIZA++ which relies on word class information, though our Back-off model does not use it. For a fair comparison, we also used IBM Model 3 as an additional baseline which does not use word class information. In addition, we used HEUR-W proposed by [27] for heuristic extraction from the last 1 sample of Back+future, and present it in +HEUR-W. Furthermore, we pruned rule tables using significance pruning [16] for a fair comparison in terms of the rule-table size denoted as +SP.

We used WMT 2006 test corpus [20] for testing, WMT 2005 test corpus [19] for tuning, and full europarl-v7 corpus for train-

[*6] Gen, Gen-Relaxed and Back took 1 day, Back+future took 1.5 days for inference on Intel Xeon E5-4650 2.70 GHz x 2 16 core 32 thread CPU with 256 MB main memory machine.

[*7] The *Sentence* denotes the sentence size and the *Word* denotes the word size for each corpus.

[*8] Our Back-off models and bayesian SCFG models do not use any word class information.

Table 3 The statistics for each corpus.

		TM	LM	Dev	Test
All	*Sentence*	1.9 M	1.9 M	2.0 k	2.0 k
De	*Word*	31.3 M	-	55.1 k	59.4 k
En	*Word*	32.8 M	53.1 M	58.8 k	55.5 k

Table 4 Results of translation evaluation in de-en full size corpus.

Model	BLEU	SIZE	
∗ GIZA++ Model 4+GDF	27.21†	73.24 M	(x15.32)
∗ GIZA++ Model 4+GDF+SP	**27.15**†	6.12 M	(x1.28)
GIZA++ Model 3+GDF	26.78	59.26 M	(x12.40)
GIZA++ Model 3+GDF+SP	26.92	5.54 M	(x1.16)
Back+future+HEUR-W	26.88	74.52 M	(x15.59)
Back+future+HEUR-W+SP	26.84	7.90 M	(x1.65)
Back+future	**27.04**	4.78 M	(x1.0)

ing. The language models were estimated from the all-English side of the europarl-v7 corpus. The statistics of these corpora are presented in **Table 3** [∗9]. The experimental set up was similar to that in Section 6.1 with the following exceptions; Slice sampling parameter a was set to 0.05. Mini-batch size was set to 1024 and sampling was performed in 20 iterations [∗10]. The translation model was extracted by the last 1 iteration. We set significance pruning threshold value as 50 used by Johnson et al. [16]. **Table 4** lists the results. The row marked up with ∗ indicates the model using word class information. We used † and bold values to separate the comparison of large and compact rule tables. The values with † represent the scores are not significantly different to the best score ($p \leq 0.05$), and the bold values indicate that the models are not significantly different to the best score ($p \leq 0.05$) in the model which rule-table sizes are less than 10 M. We used multeval [8] for significance testing. Compared to +GDF, the Heuristic extraction baselines, our Back+future can decrease the grammar size against GIZA++ with comparable BLEU score. Compared to +SP, the significantly pruned baselines, Back+future achieved statistical significantly no diffrenet BLEU score against strong baseline GIZA++ Model 4+GDF+SP on less grammar size. Because GIZA++ Model 3+GDF+SP did not achieve statistical significantly no diffrenet BLEU score against GIZA++ Model 4+GDF+SP, we can say that Back+future can outperform both BLEU score and rule-table size to the heuristically pipelined approach on same conditions. Surprisingly, HEUR-W had no gains, probably because the word alignment in each Hiero rules relied on the IBM Model 1. Even if we used +SP on HEUR-W, both the BLEU score and rule-table size are inferior to the Back+future. This result indicates that directly using a sampled tree for generating rule-tables is better than the previously used exhaustive extraction method with bayesian SCFG models, on both the BLEU score and rule-table size.

7. Analysis

Intuitively, the use of the hierarchical back-off increases the Hiero grammar size, since the phrases of all the granularities in the derivation trees are incorporated in the grammar. In contrast, our hierarchical back-off model achieved gains in translation quality without largely increasing the size of the extracted grammar when compared to the previous generative model. The major differences were the use of the minimal phrase pairs used in the previous work in which only minimal phrase pairs in the leaves of derivation trees were included in the model. As a result, larger phrase pairs were forced to be constructed from those minimal rules. On the other hand, our back-off model could directly express phrase pairs of multiple granularities. In particular, a complex noun may be composed of several Hiero rules in the previous model, but it can be directly expressed by a single phrase pair in our model. **Table 5** gives an example of a Japanese-English phrase pair which is represented by two Hiero rules in the previous model; it is directly expressed by a single phrase pair in our model. **Figure 3** shows the relationships of the generated phrase length and their sizes. Phrase lengths in these figures are calculated by adding the source and target lengths for each phrase pair. From these figures, we can see that Back and Back+future can generate longer phrase pairs covering shorter phrase pairs compared to the Gen and Gen-Relaxed in News-Commentary Spanish/French-to-English corpus and NTCIR10 Japanese-to-English corpus. However, in News-Commentary German-English corpus, there are almost no differences between Back and Gen-Relaxed. We conjecture that the lack of longer phrase pairs in Back is caused by the sparsity of German compound nouns. In NTCIR10 Japanese-to-English corpus, although the Back extracted longer phrase pairs, the BLEU score did not outperform the Gen. Japanese and English are linguistically different, and the translation of function words is decided by the context. Therefore, phrase pairs are not strongly helpful to the Japanese-to-English language pair, compared to Spanish/French-to-English language pairs.

Table 5 Example of learned rules.

Gen	*gin* X *kamera* / silver X camera *en* / salt
Back+future	*gin en kamera* / silver salt camera

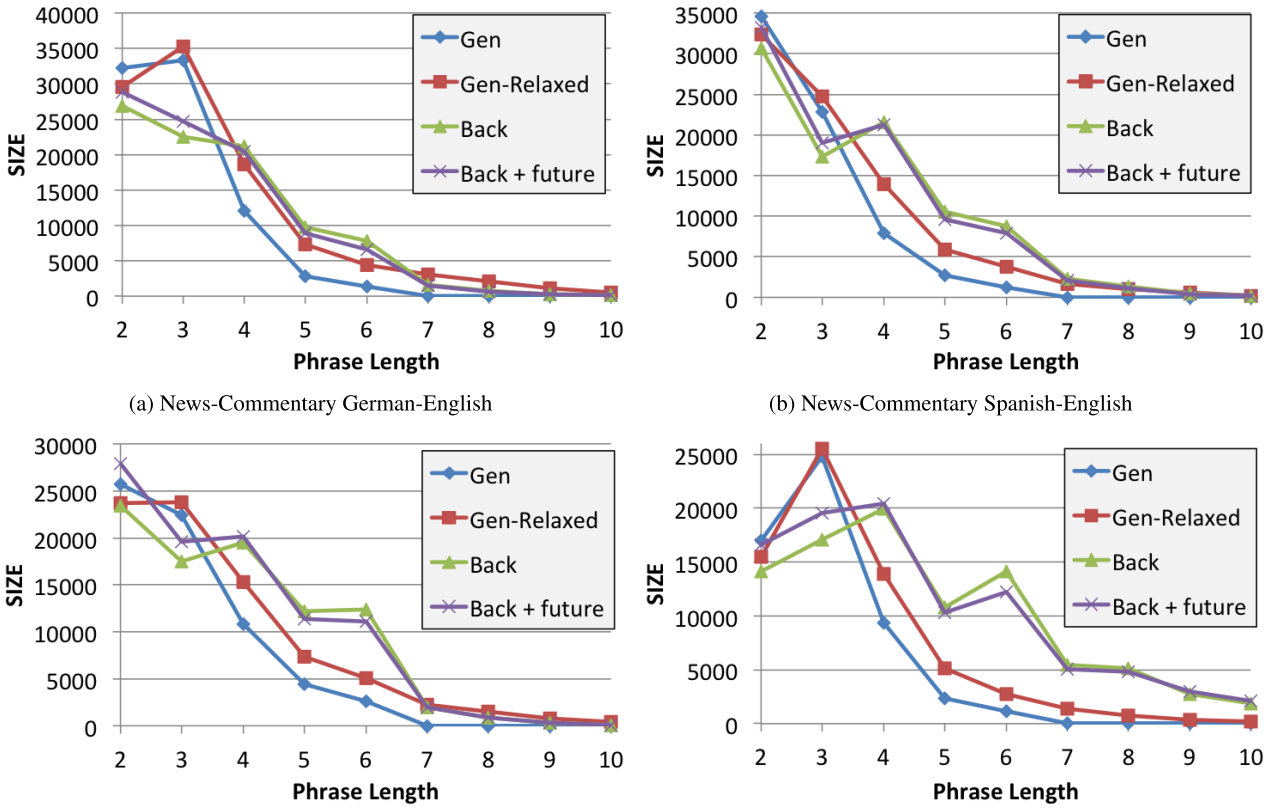
The BLEU score of Back+future was higher than the generative baseline with the comparable grammar size. **Figure 4** shows the relationships of generated rule length and their sizes [∗11]. We can observe that Back+future under generate longer rules, compared to other methods in all language pairs. We can say that Back+future generate more moderate size phrase pairs instead of longer rules, combined with the above analysis of Fig. 3. In some cases, because these longer rules increase the sparsity of rule-tables, descreasing of longer rules increases the translation quality. We can conclude that Back+future infers better models by pruning low reliable longer spans.

The BLEU score of our back-off model did not achieve gains over the heuristic baselines. The detailed analysis of the learned Hiero grammar's CRP tables reveals that the grammar is very sparse and may have little generalization capability. The expansion of back-off process and the use of word classes will solve the sparsity and increase the translation quality.

[∗9] The *Sentence* denotes the sentence size and the *Word* denotes the word size for each corpus.

[∗10] Inference took 10 days on Intel Xeon E7-8837 2.67 GHz x 4 32 core 32 thread CPU with 1,024 MB main memory machine.

[∗11] In this results, rules are limited to non-terminal rules, and have no phrase pairs.



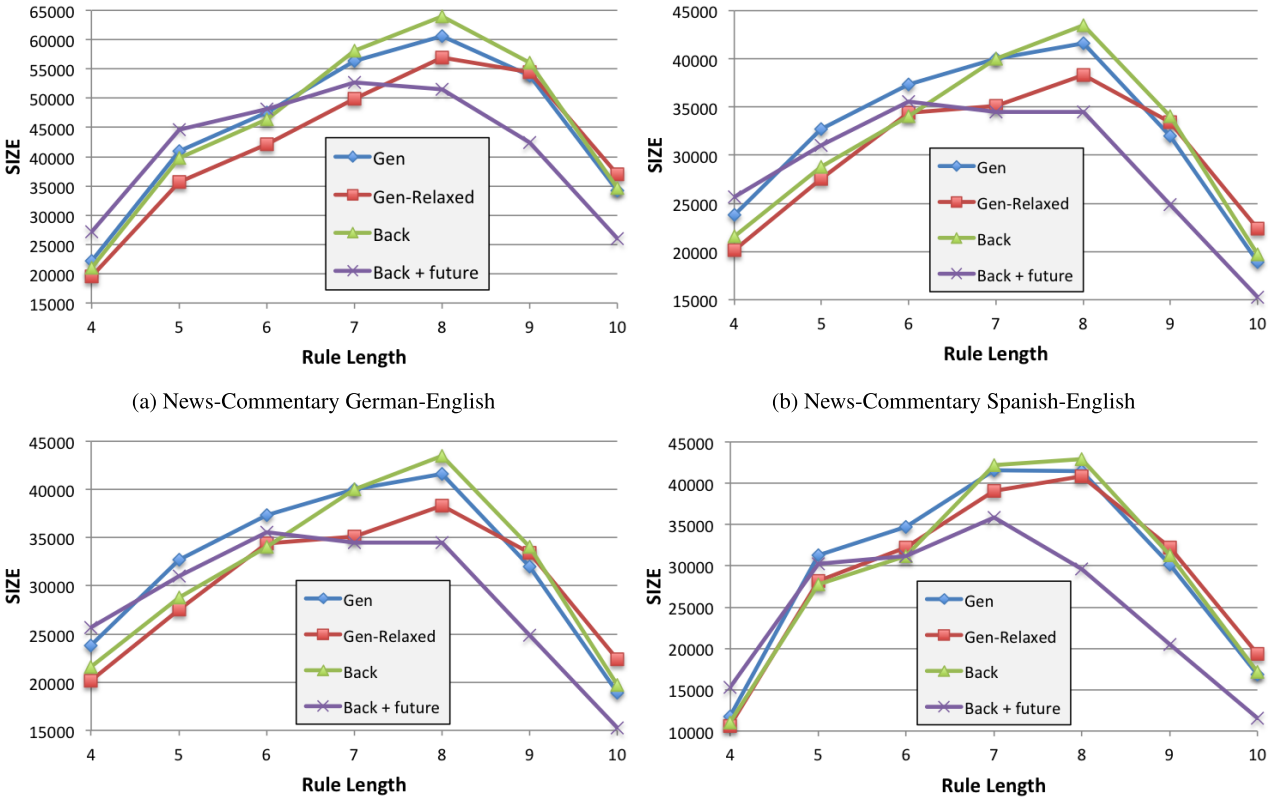
(a) News-Commentary German-English

(b) News-Commentary Spanish-English

(c) News-Commentary French-English

(d) NTCIR10 Japanese-English

Fig. 3 The relationships between length and size of phrase pairs in 100 k corpus for each model.



(a) News-Commentary German-English

(b) News-Commentary Spanish-English

(c) News-Commentary French-English

(d) NTCIR10 Japanese-English

Fig. 4 The relationships between length and size of rule pairs in 100 k corpus for each model.

8. Conclusion

We proposed a hierarchical back-off model for Hiero grammar. Our back-off model achieved higher or equal translation quality against a previous Bayesian model under BLEU scores on various language pairs: German/French/Spanish/Japanese-English. In addition to the hierarchical back-off model, we also proposed a two-step slice sampling approach. We showed that the two-step slice sampling approach can avoid over-pruning by incorporating a future score for estimating slice variables, which led to an increase in translation quality through the experiments. The joint use of the hierarchical back-off model and the two step slice sampling approach achieved comparable translation quality on a full size Germany-English language pair in Europarl v7 corpus with the significantly smaller grammar size; 10% less than that for the heuristic baseline.

For future work, we plan to embed a back-off feature to the decoder which is computed for all the phrase pairs constructed in a derivation during the decoding process. We will reflect the change of a probability as a statefull feature for the decoding step.

References

[1] Blunsom, P. and Cohn, T.: Inducing Synchronous Grammars with Slice Sampling, *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pp.238–241, Los Angeles, California, Association for Computational Linguistics (2010) (online), available from ⟨http://www.aclweb.org/anthology/N10-1028⟩.

[2] Blunsom, P., Cohn, T., Dyer, C. and Osborne, M.: A Gibbs Sampler for Phrasal Synchronous Grammar Induction, *Proc. Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP*, pp.782–790, Suntec, Singapore, Association for Computational Linguistics (2009) (online), available from ⟨http://www.aclweb.org/anthology/P/P09/P09-1088⟩.

[3] Callison-Burch, C., Koehn, P., Monz, C., Peterson, K., Przybocki, M. and Zaidan, O.F.: Findings of the 2010 Joint Workshop on Statistical Machine Translation and Metrics for Machine Translation, *Proc. Joint 5th Workshop on Statistical Machine Translation and MetricsMATR*, pp.17–53, Association for Computational Linguistics (2010).

[4] Callison-Burch, C., Koehn, P., Monz, C. and Schroeder, J.: Findings of the 2009 Workshop on Statistical Machine Translation, *Proc. 4th Workshop on Statistical Machine Translation*, Athens, Greece, pp.1–28, Association for Computational Linguistics (2009) (online), available from ⟨http://www.aclweb.org/anthology/W/W09/W09-0401⟩.

[5] Chiang, D.: Hierarchical phrase-based translation, *Computational Linguistics*, Vol.33, No.2, pp.201–228 (2007).

[6] Chiang, D.: Hope and fear for discriminative training of statistical translation models, *The Journal of Machine Learning Research*, Vol.13, No.1, pp.1159–1187 (2012).

[7] Chung, T., Fang, L., Gildea, D. and Štefankovič, D.: Sampling tree fragments from forests, *Computational Linguistics*, Vol.40, No.1, pp.203–229 (2014).

[8] Clark, J.H., Dyer, C., Lavie, A. and Smith, N.A.: Better Hypothesis Testing for Statistical Machine Translation: Controlling for Optimizer Instability, *Proc. 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pp.176–181, Portland, Oregon, USA, Association for Computational Linguistics (2011) (online), available from ⟨http://www.aclweb.org/anthology/P11-2031⟩.

[9] Cmejrek, M. and Zhou, B.: Two Methods for Extending Hierarchical Rules from the Bilingual Chart Parsing, *Coling 2010: Posters*, pp.180–188, Beijing, China, Coling 2010 Organizing Committee (2010) (online), available from ⟨http://www.aclweb.org/anthology/C10-2021⟩.

[10] Cohn, T. and Haffari, G.: An Infinite Hierarchical Bayesian Model of Phrasal Translation, *Proc. 51st Annual Meeting of the Association for Computational Linguistics* (*Volume 1: Long Papers*), Sofia, Bulgaria, pp.780–790, Association for Computational Linguistics (2013) (online), available from ⟨http://www.aclweb.org/anthology/P13-1077⟩.

[11] DeNero, J., Bouchard-Côté, A. and Klein, D.: Sampling Alignment Structure under a Bayesian Translation Model, *Proc. 2008 Conference on Empirical Methods in Natural Language Processing*, pp.314–323, Honolulu, Hawaii, Association for Computational Linguistics (2008) (online), available from ⟨http://www.aclweb.org/anthology/D08-1033⟩.

[12] DeNero, J., Gillick, D., Zhang, J. and Klein, D.: Why Generative Phrase Models Underperform Surface Heuristics, *Proc. Workshop on Statistical Machine Translation*, pp.31–38, New York City, Association for Computational Linguistics (2006) (online), available from ⟨http://www.aclweb.org/anthology/W/W06/W06-3105⟩.

[13] Dyer, C., Lopez, A., Ganitkevitch, J., Weese, J., Ture, F., Blunsom, P., Setiawan, H., Eidelman, V. and Resnik, P.: Cdec: A Decoder, Alignment, and Learning Framework for Finite-State and Context-Free Translation Models, *Proc. ACL 2010 System Demonstrations*, pp.7–12, Uppsala, Sweden, Association for Computational Linguistics (2010) (online), available from ⟨http://www.aclweb.org/anthology/P10-4002⟩.

[14] Goto, I., Chow, K.P., Lu, B., Sumita, E. and Tsou, B.K.: Overview of the patent machine translation task at the NTCIR-10 workshop, *Proc. 10th NTCIR Workshop Meeting on Evaluation of Information Access Technologies: Information Retrieval, Question Answering and Cross-Lingual Information Access, NTCIR-10* (2013).

[15] Hopkins, M. and May, J.: Tuning as Ranking, *Proc. 2011 Conference on Empirical Methods in Natural Language Processing*, Edinburgh, pp.1352–1362, Scotland, UK., Association for Computational Linguistics (2011) (online), available from ⟨http://www.aclweb.org/anthology/D11-1125⟩.

[16] Johnson, H., Martin, J., Foster, G. and Kuhn, R.: Improving Translation Quality by Discarding Most of the Phrasetable, *Proc. 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning* (*EMNLP-CoNLL*), pp.967–975, Prague, Czech Republic, Association for Computational Linguistics (2007) (online), available from ⟨http://www.aclweb.org/anthology/D/D07/D07-1103⟩.

[17] Kamigaito, H., Watanabe, T., Takamura, H., Okumura, M. and Sumita, E.: Hierarchical Back-off Modeling of Hiero Grammar based on Non-parametric Bayesian Model, *Proc. 2015 Conference on Empirical Methods in Natural Language Processing*, pp.1217–1227, Lisbon, Portugal, Association for Computational Linguistics (2015) (online), available from ⟨http://aclweb.org/anthology/D15-1143⟩.

[18] Koehn, P., Hoang, H., Birch, A., Callison-Burch, C., Federico, M., Bertoldi, N., Cowan, B., Shen, W., Moran, C., Zens, R., Dyer, C., Bojar, O., Constantin, A. and Herbst, E.: Moses: Open Source Toolkit for Statistical Machine Translation, *Proc. 45th Annual Meeting of the Association for Computational Linguistics Companion Volume Proc. Demo and Poster Sessions*, Prague, Czech Republic, pp.177–180, Association for Computational Linguistics (2007) (online), available from ⟨http://www.aclweb.org/anthology/P07-2045⟩.

[19] Koehn, P., Martin, J., Mihalcea, R., Monz, C. and Pedersen, T. (Eds.): *Proc. ACL Workshop on Building and Using Parallel Texts*, Association for Computational Linguistics, Ann Arbor, Michigan (2005).

[20] Koehn, P. and Monz, C. (Eds.): *Proceedings on the Workshop on Statistical Machine Translation*, Association for Computational Linguistics, New York City (2006).

[21] Koehn, P., Och, F.J. and Marcu, D.: Statistical phrase-based translation, *Proc. 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology-Volume 1*, pp.48–54, Association for Computational Linguistics (2003).

[22] Levenberg, A., Dyer, C. and Blunsom, P.: A Bayesian Model for Learning SCFGs with Discontiguous Rules, *Proc. 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pp.223–232, Jeju Island, Korea, Association for Computational Linguistics (2012) (online), available from ⟨http://www.aclweb.org/anthology/D12-1021⟩.

[23] Ling, W., Graça, J.A., Trancoso, I. and Black, A.: Entropy-based Pruning for Phrase-based Machine Translation, *Proc. 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pp.962–971, Jeju Island, Korea, Association for Computational Linguistics (2012) (online), available from ⟨http://www.aclweb.org/anthology/D12-1088⟩.

[24] Marcu, D. and Wong, D.: A Phrase-Based, Joint Probability Model for Statistical Machine Translation, *Proc. 2002 Conference on Empirical Methods in Natural Language Processing*, pp.133–139, Association for Computational Linguistics (online), DOI: 10.3115/1118693.1118711 (2002).

[25] Neal, R.M.: Probabilistic inference using Markov chain Monte Carlo methods (1993).

[26] Neal, R.M.: Slice sampling, *Annals of statistics*, pp.705–741 (2003).

[27] Neubig, G., Watanabe, T., Sumita, E., Mori, S. and Kawahara, T.: An Unsupervised Model for Joint Phrase Alignment and Extraction, *Proc. 49th Annual Meeting of the Association for Computational Linguis-*

tics: Human Language Technologies, pp.632–641, Portland, Oregon, USA, Association for Computational Linguistics (online), available from ⟨http://www.aclweb.org/anthology/P11-1064⟩ (2011).
[28] Och, F.J.: Minimum Error Rate Training in Statistical Machine Translation, *Proc. 41st Annual Meeting of the Association for Computational Linguistics*, pp.160–167, Sapporo, Japan, Association for Computational Linguistics (online), DOI: 10.3115/1075096.1075117 (2003).
[29] Och, F.J. and Ney, H.: A Systematic Comparison of Various Statistical Alignment Models, *Computational Linguistics*, Vol.29, No.1, pp.19–51 (2003).
[30] Papineni, K., Roukos, S., Ward, T. and Zhu, W.-J.: Bleu: A Method for Automatic Evaluation of Machine Translation, *Proc. 40th Annual Meeting of the Association for Computational Linguistics*, pp.311–318, Philadelphia, Pennsylvania, USA, Association for Computational Linguistics (online), DOI: 10.3115/1073083.1073135 (2002).
[31] Peng, X. and Gildea, D.: Type-based MCMC for Sampling Tree Fragments from Forests, *Proc. 2014 Conference on Empirical Methods in Natural Language Processing* (*EMNLP*), pp.1735–1745, Doha, Qatar, Association for Computational Linguistics (2014) (online), available from ⟨http://www.aclweb.org/anthology/D14-1180⟩.
[32] Pitman, J. and Yor, M.: The two-parameter Poisson-Dirichlet distribution derived from a stable subordinator, *The Annals of Probability*, pp.855–900 (1997).
[33] Stolcke, A. et al.: SRILM-an extensible language modeling toolkit, *Proc. International Conference on Spoken Language Processing*, pp.257–286 (2002).
[34] Teh, Y.W.: A Hierarchical Bayesian Language Model Based On Pitman-Yor Processes, *Proc. 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics*, pp.985–992, Sydney, Australia, Association for Computational Linguistics (online), DOI: 10.3115/1220175.1220299 (2006).
[35] Van Gael, J., Saatci, Y., Teh, Y.W. and Ghahramani, Z.: Beam sampling for the infinite hidden Markov model, *Proc. 25th International Conference on Machine Learning*, pp.1088–1095, ACM (2008).
[36] Wu, D.: Stochastic inversion transduction grammars and bilingual parsing of parallel corpora, *Computational linguistics*, Vol.23, No.3, pp.377–403 (1997).
[37] Xiao, X., Xiong, D., Liu, Y., Liu, Q. and Lin, S.: Unsupervised Discriminative Induction of Synchronous Grammar for Machine Translation, *Proc. COLING 2012*, pp.2883–2898, Mumbai, India, The COLING 2012 Organizing Committee (2012) (online), available from ⟨http://www.aclweb.org/anthology/C12-1176⟩.
[38] Zens, R., Stanton, D. and Xu, P.: A Systematic Comparison of Phrase Table Pruning Techniques, *Proc. 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pp.972–983, Jeju Island, Korea, Association for Computational Linguistics (2012) (online), available from ⟨http://www.aclweb.org/anthology/D12-1089⟩.
[39] Zhao, K. and Huang, L.: Minibatch and Parallelization for Online Large Margin Structured Learning, *Proc. 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp.370–379, Atlanta, Georgia, Association for Computational Linguistics (2013) (online), available from ⟨http://www.aclweb.org/anthology/N13-1038⟩.

Hidetaka Kamigaito was born in 1989. He received his B.E., M.E. and Dr.Eng. from Tokyo Institute of Technology in 2012, 2014 and 2017, respectively. He is currently a research associate at NTT Communication Science Laboratories. His research interest is machine translation and syntactic parsing.

Taro Watanebe received his B.E. and M.E. degrees in information science from Kyoto University, Kyoto, Japan in 1994 and 1997, respectively, and obtained his M.Sc. degree in language and information technologies from the School of Computer Science, Carnegie Mellon University in 2000. In 2004, he received the Ph.D. in informatics from Kyoto University, Kyoto, Japan. After working as a researcher at ATR, NTT and NICT, Dr. Watanabe is a software engineer at Google, Inc. His research interests include natural language processing, machine learning and machine translation.

Hiroya Takamura received his B.E. and M.E. from the University of Tokyo in 1997 and 2000 respectively (in 1999 he was a research student at Technische Universitaet von Wien). He received Dr.Eng. from Nara Institute of Science and Technology in 2003. He was an assistant professor at Tokyo Institute of Technology from 2003 to 2010. He is currently an associate professor at Tokyo Institute of Technology. His current research interest is computational linguistics. He is a member of the IPSJ and ACL.

Manabu Okumura was born in 1962. He received his B.E., M.E. and Dr.Eng. from Tokyo Institute of Technology in 1984, 1986 and 1989 respectively. He was an assistant at the Department of Computer Science, Tokyo Institute of Technology from 1989 to 1992, and an associate professor at the School of Information Science, Japan Advanced Institute of Science and Technology from 1992 to 2000. He is currently a professor at Institute of Innovative Research, Tokyo Institute of Technology. His current research interests include natural language processing, especially text summarization, computer assisted language learning, sentiment analysis, and text data mining.

Eiichiro Sumita received his PhD in Engineering from Kyoto University in 1999, and a Master's and Bachelor's in Computer Science from the University of Electro-Communications in 1982 and 1980, respectively. He is now in the National Institute of Information and Communication Technology (NICT), its fellow and the associate director-general of Advanced Speech Translation Research and Development Promotion Center (ASTREC). His research interests cover Machine Translation and e-Learning. He is a co-recipient of the Maejima Hisoka Prize in 2013, the Commendation for Science and Technology by the Minister of Education, Culture, Sports, Science and Technology, Prizes for Science and Technology in 2010, and the AAMT Nagao Award in 2007.