Hierarchical Phrase-based Translation Grammars Extracted from Alignment Posterior Probabilities

Adrià de Gispert, Juan Pino, William Byrne
Machine Intelligence Laboratory
Department of Engineering, University of Cambridge
Trumpington Street, CB2 1PZ, U.K.
{ad465|jmp84|wjb31}@eng.cam.ac.uk

Abstract

We report on investigations into hierarchical phrase-based translation grammars based on rules extracted from posterior distributions over alignments of the parallel text. Rather than restrict rule extraction to a single alignment, such as Viterbi, we instead extract rules based on posterior distributions provided by the HMM word-to-word alignment model. We define translation grammars progressively by adding classes of rules to a basic phrase-based system. We assess these grammars in terms of their expressive power, measured by their ability to align the parallel text from which their rules are extracted, and the quality of the translations they yield. In Chinese-to-English translation, we find that rule extraction from posteriors gives translation improvements. We also find that grammars with rules with only one nonterminal, when extracted from posteriors, can outperform more complex grammars extracted from Viterbi alignments. Finally, we show that the best way to exploit source-to-target and target-to-source alignment models is to build two separate systems and combine their output translation lattices.

1 Introduction

Current practice in hierarchical phrase-based translation extracts regular phrases and hierarchical rules from word-aligned parallel text. Alignment models estimated over the parallel text are used to generate these alignments, but these models are then typically used no further in rule extraction. This is less than ideal because these alignment models, even if they are not suitable for direct use in translation, can still provide a great deal of useful information beyond a single best estimate of the alignment of the parallel text. Our aim is to use alignment models to generate the statistics needed to build translation grammars. The challenge in doing so is to extend the current procedures, which are geared towards the use of a single alignment, to make more of what can be provided by alignment models. The goal is to extract a richer and more robust set of translation rules.

There are two aspects to hierarchical phrase-based translation grammars which concern us. The first is expressive power, which we take as the ability to generate known reference translations from sentences in the source language. This is determined by the degree of phrase movements and the translations allowed by the rules of the grammar. For a grammar with given types of rules, larger rule sets will yield greater expressive power. This motivates studies of grammars based on the rules which are extracted and the movement the grammar allows. The second aspect is of course translation accuracy. If the expressive power is adequate, then the desire is that the grammar assigns a high score to a correct translation.

We use posterior probabilities over parallel data to address both of these aspects. These posteriors allow us to build larger rule sets with improved translation accuracy. Ideally, for a sentence pair we wish to consider all possible alignments between all possible source and target phrases within these sentences. Given a grammar allowing certain types of movement, we would then extract all possible parses that are consistent with any alignments of these phrases.

Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing, pages 545–554,
MIT, Massachusetts, USA, 9-11 October 2010.

To make this approach feasible, we consider only phrase-to-phrase alignments with a high posterior probability under the alignment models. In this way, the alignment model probabilities guide rule extraction.

The paper is organized as follows. Section 2 reviews related work on using posteriors to extract phrases, as well as other approaches that tightly integrate word alignment and rule extraction. Section 3 describes rule extraction based on word and phrase posterior distributions provided by the HMM word-to-word alignment model. In Section 4 we define translation grammars progressively by adding classes of rules to a basic phrase-based system, motivating each rule type by the phrase movement it is intended to achieve. In Section 5 we assess these grammars in terms of their expressive power and the quality of the translations they yield in Chinese-to-English, showing that rule extraction from posteriors gives translation improvements. We also find that the best way to exploit source-to-target and target-to-source alignment models is to build two separate systems and combine their output translation lattices. Section 6 presents the main conclusions of this work.

2 Related Work

Some authors have previously addressed the limitation caused by decoupling word alignment models from grammar extraction. For instance Venugopal et al. (2008) extract rules from n-best lists of alignments for a syntax-augmented hierarchical system. Alignment n-best lists are also used in Liu et al. (2009) to create a structure called weighted alignment matrices that approximates word-to-word link posterior probabilities, from which phrases are extracted for a phrase-based system. Alignment posteriors have been used before for extracting phrases in non-hierarchical phrase-based translation (Venugopal et al., 2003; Kumar et al., 2007; Deng and Byrne, 2008).

In order to simplify hierarchical phrase-based grammars and make translation feasible with relatively large parallel corpora, some authors discuss the need for various filters during rule extraction (Chiang, 2007). In particular Lopez (2008) enforces a minimum span of two words per nonterminal, Zollmann et al. (2008) use a minimum count threshold for all rules, and Iglesias et al. (2009) propose a finer-grained filtering strategy based on rule patterns. Other approaches include insisting that target-side rules are well-formed dependency trees (Shen et al., 2008).

We also note approaches to tighter coupling between translation grammars and alignments. Marcu and Wong (2002) describe a joint-probability phrase-based model for alignment, but the approach is limited due to excessive complexity as Viterbi inference becomes NP-hard (DeNero and Klein, 2008). More recently, Saers et al. (2009) report improvement on a phrase-based system where word alignment has been trained with an inversion transduction grammar (ITG) rather than IBM models. Pauls et al. (2010) also use an ITG to directly align phrases to nodes in a string-to-tree model. Bayesian methods have been recently developed to induce a grammar directly from an unaligned parallel corpus (Blunsom et al., 2008; Blunsom et al., 2009). Finally, Cmejrek et al. (2009) extract rules directly from bilingual chart parses of the parallel corpus without using word alignments. We take a different approach in that we aim to start with very strong word alignment models and use them to guide grammar extraction.

3 Rule Extraction from Alignment Posteriors

The goal of rule extraction is to generate a set of good-quality translation rules from a parallel corpus. Rules are of the form $X \rightarrow \langle \gamma, \alpha, \sim \rangle$, where $\gamma, \alpha \in \{X \cup \mathbf{T}\}^+$ are the source and target sides of the rule, $\mathbf{T}$ denotes the set of terminals (words) and $\sim$ is a bijective function[1] relating source and target nonterminals X of each rule (Chiang, 2007). For each γ, the probability over translations α is set by relative frequency over the extracted examples from the corpus.

We take a general approach to rule extraction, as described by the following procedure. For simplicity we discuss the extraction of regular phrases, that is, rules of the form $X \rightarrow \langle w, w \rangle$, where $w \in \{\mathbf{T}\}^+$. Section 3.3 extends this procedure to rules with non-

[1] This function is defined if there are at least two nonterminals, and for clarity of presentation will be omitted in this paper

terminal symbols.

Given a sentence pair (f_1^J, e_1^I), the extraction algorithm traverses the source sentence and, for each sequence of terminals $f_{j_1}^{j_2}$, it considers all possible target-side sequences $e_{i_1}^{i_2}$ as translation candidates. Each target-side sequence that satisfies the alignment constraints $\mathcal{C}_A$ is ranked by the function f_R. For practical reasons, a set of selection criteria $\mathcal{C}_S$ is then applied to these ranked candidates and defines the set of translations of the source sequence that are extracted as rules. Each extracted rule is assigned a count f_C.

In this section we will explore variations of this rule extraction procedure involving alternative definitions of the ranking and counting functions, f_R and f_C, based on probabilities over alignment models.

Common practice (Koehn et al., 2003) takes a set of word alignment links $\mathbf{L}$ and defines the alignment constraints $\mathcal{C}_A$ so that there is a *consistency* between the links in the $(f_{j_1}^{j_2}, e_{i_1}^{i_2})$ phrase pair. This is expressed by $\forall(j,i) \in \mathbf{L} : (j \in [j_1, j_2] \wedge i \in [i_1, i_2]) \vee (j \notin [j_1, j_2] \wedge i \notin [i_1, i_2])$. If these constraints are met, then alignment probabilities are ignored and $f_R = f_C = 1$. We call this extraction Viterbi-based, as the set of alignment links is generally obtained after applying a symmetrization heuristic to source-to-target and target-to-source Viterbi alignments.

In the following section we depart from this approach and apply novel functions to rank and count target-side translations according to their quality in the context of each parallel sentence, as defined by the word alignment models. We also depart from common practice in that we do not use a set of links as alignment constraints. We thus find an increase in the number of extracted rules, and consequently better relative frequency estimates over translations.

3.1 Ranking and Counting Functions

We describe two alternative approaches to modify the functions f_R and f_C so that they incorporate the probabilities provided by the alignment models.

3.1.1 Word-to-word Alignment Posterior Probabilities

Word-to-word alignment posterior probabilities $p(l_{ji}|f_1^J, e_1^I)$ express how likely it is that the words in source position j and target position i are aligned given a sentence pair. These posteriors can be efficiently computed for Model 1, Model 2 and HMM, as described in (Brown et al., 1993; Venugopal et al., 2003; Deng and Byrne, 2008).

We will use these posteriors in functions to score phrase pairs. For a simple non-disjoint case $(f_{j_1}^{j_2}, e_{i_1}^{i_2})$ we use:

$$f_R(f_{j_1}^{j_2}, e_{i_1}^{i_2}) = \prod_{j=j_1}^{j_2} \sum_{i=i_1}^{i_2} \frac{p(l_{ji}|f_1^J, e_1^I)}{i_2 - i_1 + 1} \qquad (1)$$

which is very similar to the score used for lexical features in many systems (Koehn, 2010), with the link posteriors for the sentence pair playing the role of the Model 1 translation table.

For a particular source phrase, Equation 1 is not a proper conditional probability distribution over all phrases in the target sentence. Therefore it cannot be used as such without further normalization. Indeed we find that this distribution is too sharp and overemphasises short phrases, so we use $f_C = 1$. However, it does allow us to rank target phrases as possible translations. In contrast to the common extraction procedure described in the previous section, the ranking approach described here can lead to a much more exhaustive extraction unless selection criteria are applied. These we describe in Section 3.2.

We note that Equation 1 can be computed using link posteriors provided by alignment models trained on either source-to-target or target-to-source translation directions.

3.1.2 Phrase-to-phrase Alignment Posterior Probabilities

Rather than limit ourselves to word-to-word link posteriors we can define alignment probability distributions over phrase alignments. We do this by defining the set of alignments A as $A(j_1, j_2; i_1, i_2) = \{a_1^J : a_j \in [i_1, i_2] \text{ iff } j \in [j_1, j_2]\}$, where a_j is the random process that describes word-to-word alignments. These are the alignments from which the phrase pair $(f_{j_1}^{j_2}, e_{i_1}^{i_2})$ would be extracted.

The posterior probability of these alignments given the sentence pair is defined as follows:

$$p(A|e_1^I, f_1^J) = \frac{\sum_{a_1^J \in A} p(f_1^J, a_1^J|e_1^I)}{\sum_{a_1^J} p(f_1^J, a_1^J|e_1^I)} \qquad (2)$$

G_0	G_1	G_2	G_3
$S \rightarrow \langle X, X \rangle$	$X \rightarrow \langle w\ X, X\ w \rangle$	$X \rightarrow \langle w\ X, X\ w \rangle$	$X \rightarrow \langle w\ X, X\ w \rangle$
$S \rightarrow \langle S\ X, S\ X \rangle$	$X \rightarrow \langle X\ w, w\ X \rangle$	$X \rightarrow \langle X\ w, w\ X \rangle$	$X \rightarrow \langle X\ w, w\ X \rangle$
$X \rightarrow \langle w, w \rangle$		$X \rightarrow \langle w\ X, w\ X \rangle$	$X \rightarrow \langle w\ X, w\ X \rangle$
			$X \rightarrow \langle w\ X\ w, w\ X\ w \rangle$

Table 1: Hierarchical phrase-based grammars containing different types of rules. The grammar expressivity is greater as more types of rules are included. In addition to the rules shown in the respective columns, G_1, G_2 and G_3 also contain the rules of G_0.

With IBM models 1 and 2, the numerator and denominator in Equation 2 can be computed in terms of posterior link probabilities (Deng, 2005). With the HMM model, the denominator is computed using the forward algorithm while the numerator can be computed using a modified forward algorithm (Deng, 2005).

These phrase posteriors directly define a probability distribution over the alignments of translation candidates, so we use them both for ranking and scoring extracted rules, that is $f_R = f_C = p$. This approach assigns a fractional count to each extracted rule, which allows finer estimation of the forward and backward translation probability distributions.

3.2 Alignment Constraints and Selection Criteria

In order to keep this process computationally tractable, some extraction constraints are needed. In order to extract a phrase pair $(f_{j_1}^{j_2}, e_{i_1}^{i_2})$, we define the following:

- $\mathcal{C}_A$ requires at least one pair of positions $(j, i) : (j \in [j_1, j_2] \wedge i \in [i_1, i_2])$ with word-to-word link posterior probability $p(l_{ji}|f_1^J, e_1^I) > 0.5$, and that there is no pair of positions $(j, i) : (j \in [j_1, j_2] \wedge i \notin [i_1, i_2]) \vee (j \notin [j_1, j_2] \wedge i \in [i_1, i_2])$ with $p(l_{ji}|f_1^J, e_1^I) > 0.5$

- $\mathcal{C}_S$ allows only the k best translation candidates to be extracted. We use $k = 3$ for regular phrases, and $k = 2$ for hierarchical rules.

Note that we do not discard rules according to their scores f_C at this point (unlike Liu et al. (2009)), since we prefer to add all phrases from all sentence pairs before carrying out such filtering steps.

Once all rules over the entire collection of parallel sentences have been extracted, we require each rule to occur at least n_{obs} times and with a forward translation probability $p(\alpha|\gamma) > 0.01$ to be used for translation.

3.3 Extraction of Rules with Nonterminals

Extending the procedure previously described to the case of more complex hierarchical rules including one or even two nonterminals is conceptually straightforward. It merely requires that we traverse the source and target sentences and consider possibly disjoint phrase pairs. Optionally, the alignment constraints can also be extended to apply on the nonterminal X.

Equation 1 is then only modified in the limits of the product and summation, whereas Equation 2 remains unchanged, as long as the set of valid alignments A is redefined. For example, for a rule of the form $X \rightarrow \langle w\ X\ w, w\ X\ w \rangle$, we use $A \equiv A(j_1, j_2; j_3, j_4; i_1, i_2; i_3, i_4)$.

4 Hierarchical Translation Grammar Definition

In this section we define the hierarchical phrase-based synchronous grammars we use for translation experiments. Each grammar is defined by the type of hierarchical rules it contains. The rule type can be obtained by replacing every sequence of terminals by a single symbol 'w', thus ignoring the identity of the words, but capturing its generalized structure and the kind of reordering it encodes (this was defined as rule pattern in Iglesias et al. (2009)).

A monotonic phrase-based translation grammar G_0 can be defined as shown in the left-most column of Table 1; it includes all regular phrases, represented by the rule type $X \rightarrow \langle w, w \rangle$, and the two glue

(G_0) R^1: $S \rightarrow \langle X, X \rangle$
(G_0) R^2: $X \rightarrow \langle \mathrm{s}_2\ \mathrm{s}_3, \mathrm{t}_2 \rangle$
(G_1) R^3: $X \rightarrow \langle \mathrm{s}_1\ X, X\ \mathrm{t}_3 \rangle$
(G_1) R^4: $X \rightarrow \langle X\ \mathrm{s}_4, \mathrm{t}_1\ X \rangle$
(G_2) R^5: $X \rightarrow \langle \mathrm{s}_1\ X, \mathrm{t}_7\ X \rangle$
(G_3) R^6: $X \rightarrow \langle \mathrm{s}_1\ X\ \mathrm{s}_4, \mathrm{t}_5\ X\ \mathrm{t}_6 \rangle$

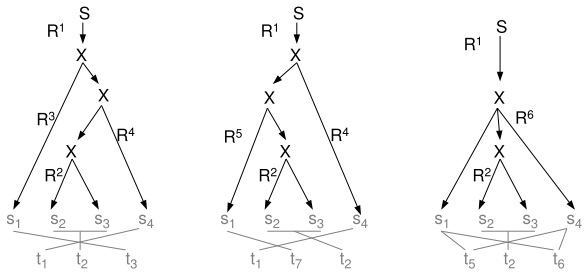

Figure 1: Example of a hierarchical translation grammar and two parsing trees following alternative rule derivations for the input sentence $s_1 s_2 s_3 s_4$.

rules that allow concatenation. Our approach is now simple: we extend this grammar by successively incorporating sets of hierarchical rules. The goal is to obtain a grammar with few rule types but which is capable of generating a rich set of translation candidates for a given input sentence.

With this in mind, we define the following three grammars, also summarized in Table 1:

- $G_1 := G_0 \bigcup$ $\{\ X \rightarrow \langle w\ X, X\ w \rangle\ ,\ X \rightarrow \langle X\ w, w\ X \rangle\ \}$. This incorporates reordering capabilities with two rule types that place the unique nonterminal in an opposite position in each language; we call these 'phrase swap rules'. Since all nonterminals are of the same category X, nested reordering is possible. However, this needs to happen consecutively, *i.e.* a swap must apply after a swap, or the rule is concatenated with the glue rule.

- $G_2 := G_1 \bigcup \{\ X \rightarrow \langle w\ X, w\ X \rangle\ \}$. This adds monotonic concatenation capabilities to the previous translation grammar. The glue rule already allows rule concatenation. However, it does so at the S category, that is, it concatenates phrases and rules *after* they have been reordered, in order to complete a sentence. With this new rule type, G_2 allows phrase/rule concatenation *before* reordering with another hierarchical rule. Therefore, nested reordering does not require successive swaps anymore.

- $G_3 := G_2 \bigcup \{\ X \rightarrow \langle w\ X\ w, w\ X\ w \rangle\ \}$. This adds single nonterminal rules with disjoint terminal sequences, which can encode a monotonic or reordered relationship between them, depending on what their alignment was in the parallel corpus. Although one could expect the movement captured by this phrase-disjoint rule type to be also present in G_2 (via two swaps or one concatenation plus one swap), the terminal sequences w may differ.

Figure 1 shows an example set of rules indicating to which of the previous grammars each rule belongs, and shows three translation candidates as generated by grammars G_1 (left-most tree), G_2 (middle tree) and G_3 (right-most tree). Note that the middle tree cannot be generated with G_1 as it requires monotonic concatenation before reordering with rule R^4.

The more rule types a hierarchical grammar contains, the more different rule derivations and the greater the search space of alternative translation candidates. This is also connected to how many rules are extracted per rule type. Ideally we would like the grammar to be able to generate the correct translation of a given input sentence, without overgenerating too many other candidates, as that makes the translation task more difficult.

We will make use of the parallel data in measuring the ability of a grammar to generate correct translations. By extracting rules from a parallel sentence, we translate them and observe whether the translation grammar is able to produce the parallel target translation. In Section 5.1 we evaluate this for a Chinese-to-English task.

4.1 Reducing Grammar Redundancy

Let us discuss grammar G_2 in more detail. As described in the previous section, the motivation for including rule type $X \rightarrow \langle w\ X, w\ X \rangle$ is that the grammar be able to carry out monotonic concatenation *before* applying another hierarchical rule with reordering. This movement is permitted by this rule type, but the use of a single nonterminal category X also allows the grammar to apply the concatenation *after* reordering, that is, immediately before the glue rule is applied. This creates significant redundancy in rule derivations, as this rule type is allowed to act as a glue rule. For example, given an input sentence $s_1 s_2$ and the following simple grammar:

$$\begin{aligned}
&\mathrm{R}^0\text{: } S \rightarrow \langle X, X \rangle \\
&\mathrm{R}^1\text{: } S \rightarrow \langle S\ X, S\ X \rangle \\
&\mathrm{R}^2\text{: } X \rightarrow \langle \mathrm{s}_1, \mathrm{t}_1 \rangle \\
&\mathrm{R}^3\text{: } X \rightarrow \langle \mathrm{s}_2, \mathrm{t}_2 \rangle \\
&\mathrm{R}^4\text{: } X \rightarrow \langle \mathrm{s}_1\ X, \mathrm{t}_1\ X \rangle
\end{aligned}$$

two derivations are possible: R^2,R^0,R^3,R^1 and R^3,R^4,R^0, and the translation result is identical.

To avoid this situation we introduce a nonterminal M in the left-hand side of monotonic concatenation rules of G_2. All rules are allowed to use nonterminals X and M in their right-hand side, except the glue rules, which can only take X. In the context of our example, R^4 is substituted by:

$$\begin{aligned}
&\mathrm{R}^{4a}\text{: } M \rightarrow \langle \mathrm{s}_1\ X, \mathrm{t}_1\ X \rangle \\
&\mathrm{R}^{4b}\text{: } M \rightarrow \langle \mathrm{s}_1\ M, \mathrm{t}_1\ M \rangle
\end{aligned}$$

so that only the first derivation is possible: R^2,R^0,R^3,R^1, because applying R^3,R^{4a} yields a nonterminal M that cannot be taken by the glue rule R^0.

5 Experiments

We report experiments in Chinese-to-English translation. Our system is trained on a subset of the GALE 2008 evaluation parallel text;[2] this is approximately 50M words per language. We report translation results on a development set *tune-nw* and a test set *test-nw1*. These contain translations produced by the GALE program and portions of the newswire sections of MT02 through MT06. They contain 1,755 sentences and 1,671 sentences respectively. Results are also reported on a smaller held-out test set *test-nw2*, containing 60% of the NIST newswire portion of MT06, that is, 369 sentences.

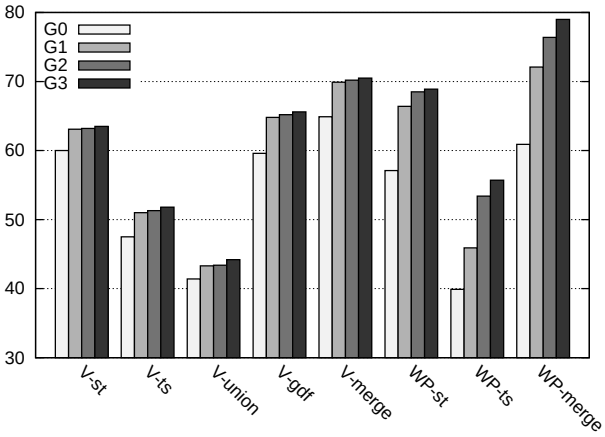

Figure 2: Percentage of parallel sentences successfully aligned for various extraction methods and grammars.

The parallel texts for both language pairs are aligned using MTTK (Deng and Byrne, 2008). For decoding we use HiFST, a lattice-based decoder implemented with Weighted Finite State Transducers (de Gispert et al., 2010). Likelihood-based search pruning is applied if the number of states in the lattice associated with each CYK grid cell exceeds 10,000, otherwise the entire search space is explored. The language model is a 4-gram language model estimated over the English side of the parallel text and the AFP and Xinhua portions of the English Gigaword Fourth Edition (LDC2009T13), interpolated with a zero-cutoff stupid-backoff (Brants et al., 2007) 5-gram estimated using 6.6B words of English newswire text. In tuning the systems, standard MERT (Och, 2003) iterative parameter estimation under IBM BLEU[3] is performed on the development sets.

5.1 Measuring Expressive Power

We measure the expressive power of the grammars described in the previous section by running the translation system in alignment mode (de Gispert et al., 2010) over the parallel corpus. Conceptually, this is equivalent to replacing the language model by the target sentence and seeing if the system is able to find any candidate. Here the weights assigned to the

[2] See http://projects.ldc.upenn.edu/gale/data/catalog.html. We excluded the UN material and the LDC2002E18, LDC2004T08, LDC2007E08 and CUDonga collections.

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

Grammar	Extraction	# Rules	*tune-nw*			*test-nw1*	*test-nw2*
			time	*prune*	BLEU	BLEU	BLEU
G_H	**V-union**	979149	3.7	0.3	35.1	35.6	37.6
G_1	**V-union**	613962	0.4	0.0	33.6	34.6	36.4
	WP-st	920183	0.9	0.0	34.3	34.8	37.5
	PP-st	893542	1.4	0.0	34.4	35.1	37.7
G_2	**V-union**	734994	1.0	0.0	34.5	35.4	37.2
	WP-st	1132386	5.8	0.5	35.1	36.0	37.7
	PP-st	1238235	7.8	0.7	35.5	36.4	38.2
G_3	**V-union**	966828	1.2	0.0	34.9	35.3	37.0
	WP-st	2680712	8.3	1.1	35.1	36.2	37.9
	PP-st	5002168	10.7	2.6	35.5	36.4	38.5

Table 2: Chinese-to-English translation results with alternative grammars and extraction methods (lower-cased BLEU shown). Time (secs/word) and prune (times/word) measurements done on *tune-nw* set.

rules are irrelevant, as only the ability of the grammar to create a desired hypothesis is important.

We compare the percentage of target sentences that can be successfully produced by grammars G_0, G_1, G_2 and G_3 for the following extraction methods:

- **Viterbi (V)**. This is the standard extraction method based on a set of alignment links. We distinguish four cases, depending on the model used to obtain the set of links: source-to-target (**V-st**), target-to-source (**V-ts**), and two common symmetrization strategies: union (**V-union**) and grow-diag-final (**V-gdf**), described in (Koehn et al., 2003).

- **Word Posteriors (WP)**. The extraction method is based on word alignment posteriors described in Section 3.1.1. These rules can be obtained either from the posteriors of the source-to-target (**WP-st**) or the target-to-source (**WP-ts**) alignment models. We apply the alignment constraints and selection criteria described in Section 3.2. We do not report alignment percentages when using phrase posteriors (as described in Section 3.1.2) as they are roughly identical to the **WP** case.

- Finally, in both cases, we also report results when merging the extracted rules in both directions into a single rule set (**V-merge** and **WP-merge**).

Figure 2 shows the results obtained for a random selection of 10,000 parallel corpus sentences. As expected, we can see that for any extraction method, the percentage of aligned sentences increases when switching from G_0 to G_1, G_2 and G_3. Posterior-based extraction is shown to outperform standard methods based on a Viterbi set of alignments for nearly all grammars. The highest alignment percentages are obtained when merging rules obtained under models trained in each direction (**WP-merge**), approximately reaching 80% for grammar G_3.

The maximum rule span in alignment was allowed to be 15 words, so as to be similar to translation, where the maximum rule span is 10 words. Relaxing this in alignment to 30 words yields approximately 90% coverage for **WP-merge** under G_3.

We note that if alignment constraints $\mathcal{C}_A$ and selection criteria $\mathcal{C}_S$ were not applied, that is $k = \infty$, then alignment percentages would be 100% even for G_0, but the extracted grammar would include many noisy rules with poor generalization power and would suffer from overgeneration.

5.2 Translation Results

In this section we investigate the translation performance of each hierarchical grammar, as defined by rules obtained from three rule extraction methods:

- **Viterbi union (V-union)**. Standard rule extraction from the union of the source-to-target and target-to-source alignment link sets.

- **Word Posteriors (WP-st)**. Extraction based on word posteriors as described in Section 3.1.1. The posteriors are provided by the source-to-target alignment model. Alignment constraints and selection criteria of Section 3.2 are applied, with $n_{obs} = 2$.

- **Phrase Posteriors (PP-st)**. Extraction based on phrase alignment posteriors, as described in Section 3.1.2, with fractional counts proportional to the phrase probability under the source-to-target alignment model. Alignment constraints and selection criteria of Section 3.2 are applied, with $n_{obs} = 0.2$.

Table 2 reports the translation results, as well as the number of extracted rules in each case. It also shows the following decoding statistics as measured on the *tune-nw* set: decoding time in seconds per input word, and number of instances of search pruning (described in Section 5) per input word.

As a contrast, we extract rules according to the heuristics introduced in (Chiang, 2007) and apply the filters described in (Iglesias et al., 2009) to generate a standard hierarchical phrase-based grammar G_H. This uses rules with up to two nonadjacent nonterminals, but excludes identical rule types such as $X \rightarrow \langle w\ X, w\ X \rangle$ or $X \rightarrow \langle w\ X_1\ w\ X_2, w\ X_1\ w\ X_2 \rangle$, which were reported to cause computational difficulties without a clear improvement in translation (Iglesias et al., 2009).

Grammar expressivity. As expected, for the standard extraction method (see rows entitled **V-union**), grammar G_1 is shown to underperform all other grammars due to its structural limitations. On the other hand, grammar G_2 obtains much better scores, nearly generating the same translation quality as the baseline grammar G_H. Finally, G_3 does not prove able to outperform G_2, which suggests that the phrase-disjoint rules with one nonterminal are redundant for the translation grammar.

Rule extraction method. For all grammars, we find that the proposed extraction methods based on alignment posteriors outperform standard Viterbi-based extraction, with improvements ranging from 0.5 to 1.1 BLEU points for *test-nw1* (depending on the grammar) and from 1.0 to 1.5 for *test-nw2*. In all cases, the use of phrase posteriors **PP** is the best option. Interestingly, we find that G_2 extracted with **WP** and **PP** methods outperforms the more complex G_H grammar as obtained from Viterbi alignments.

Rule set statistics. For grammar G_2 evaluated on the *tune-nw* set, standard Viterbi-based extraction produces 0.7M rules, whereas the WP and PP extraction methods yield 1.1M and 1.2M rules respectively. We further analyse the sets of rules $X \rightarrow \langle \gamma, \alpha, \sim \rangle$ in terms of the number of distinct source and target sequences γ and α which are extracted. Viterbi extraction yields 82k distinct source sequences whereas the WP and PP methods yield 116k and 146k sequences, respectively. In terms of the average number of target sequences for each source sequence, Viterbi extraction yields an average of 8.7 while WP and PP yield 9.7 and 8.4 rules on average. This shows that method **PP** yields wider coverage but with sharper forward rule translation probability distributions than method **WP**, as the average number of translations per rule is determined by the $p(\alpha|\gamma) > 0.01$ threshold mentioned in Section 3.2.

Decoding time and pruning in search. In connection to the previous comments, we find an increased need for search pruning, and subsequently slower decoding speed, as the search space grows larger with methods **WP** and **PP**. A larger search space is created by the larger rule sets, which allows the system to generate new hypotheses of better quality.

5.3 Rule Concatenation in Grammar G_2

In Section 4.1 we described a strategy to reduce grammar redundancy by introducing an additional nonterminal M for monotonic concatenation rules. We find that without this distinction among nonterminals, search pruning and decoding time are increased by a factor of 1.5, and there is a slight degradation in BLEU ($\sim$0.2) as more search errors are introduced.

Another relevant aspect of this grammar is the actual rule type selected for monotonic concatenation. We described using type $X \rightarrow \langle w\ X, w\ X \rangle$ (concatenation on the right), but one could also include $X \rightarrow \langle X\ w, X\ w \rangle$ (concatenation on the left), or both, for the same purpose. We evaluated the three alternatives and found that scores are identical when either including right or left concatenation types, but including both is harmful for performance, as the need to prune and decoding time increase by a fac-

tor of 5 and 4, respectively, and we observe again a slight degradation in performance.

Rule Extraction	*tune-nw*	*test-nw1*	*test-nw2*
V-st	34.7	35.6	37.5
V-ts	34.0	34.8	36.6
V-union	34.5	35.4	37.2
V-gdf	34.4	35.3	37.1
WP-st	35.1	36.0	37.7
WP-ts	34.5	35.0	37.0
PP-st	35.5	36.4	38.2
PP-ts	34.8	35.3	37.2
PP-merge	35.5	36.4	38.4
PP-merge-MERT	35.5	36.4	38.3
LMBR**(V-st)**	35.0	35.8	38.4
LMBR**(V-st,V-ts)**	35.5	36.3	38.9
LMBR**(PP-st)**	36.1	36.8	38.8
LMBR**(PP-st,PP-ts)**	36.4	36.9	39.3

Table 3: Translation results under grammar G_2 with individual rule sets, merged rule sets, and rescoring and system combination with lattice-based MBR (lower-cased BLEU shown)

5.4 Symmetrizing Alignments of Parallel Text

In this section we investigate extraction from alignments (and posterior distributions) over parallel text which are generated using alignment models trained in the source-to-target (**st**) and target-to-source (**ts**) directions. Our motivation is that symmetrization strategies have been reported to be beneficial for Viterbi extraction methods (Och and Ney, 2003; Koehn et al., 2003).

Results are shown in Table 3 for grammar G_2. We find that rules extracted under the source-to-target alignment models (**V-st**, **WP-st** and **PP-st**) consistently perform better than the **V-ts**, **WP-ts** and **PP-ts** cases. Also, for Viterbi extraction we find that the source-to-target **V-st** case performs better than any of the symmetrization strategies, which contradicts previous findings for non-hierarchical phrase-based systems(Koehn et al., 2003).

We use the **PP** rule extraction method to extract two sets of rules, under the **st** and **ts** alignment models respectively. We now investigate two ways of merging these sets into a single grammar for translation. The first strategy is **PP-merge** and merges both rule sets by assigning to each rule the maximum count assigned by either alignment model. We then extend the previous strategy by adding three binary feature functions to the system, indicating whether the rule was extracted under the '**st**' model, the '**ts**' model or both. The motivation is that MERT can weight rules differently according to the alignment model they were extracted from. However, we do not find any improvement with either strategy.

Finally, we use linearised lattice minimum Bayes-risk decoding (Tromble et al., 2008; Blackwood et al., 2010) to combine translation lattices (de Gispert et al., 2010) as produced by rules extracted under each alignment direction (see rows named LMBR**(V-st,V-ts)** and LMBR**(PP-st,PP-ts)**). Gains are consistent when comparing this to applying LMBR to each of the best individual systems (rows named LMBR**(V-st)** and LMBR**(PP-st)**). Overall, the best-performing strategy is to extract two sets of translation rules under the phrase pair posteriors in each translation direction, and then to perform translation twice and merge the results.

6 Conclusion

Rule extraction based on alignment posterior probabilities can generate larger rule sets. This results in grammars with more expressive power, as measured by the ability to align parallel sentences. Assigning counts equal to phrase posteriors produces better estimation of rule translation probabilities. This results in improved translation scores as the search space grows.

This more exhaustive rule extraction method permits a grammar simplification, as expressed by the phrase movement allowed by its rules. In particular a simple grammar with rules of only one nonterminal is shown to outperform a more complex grammar built on rules extracted from Viterbi alignments. Finally, we find that the best way to exploit alignment models trained in each translation direction is to extract two rule sets based on alignment posteriors, translate the input independently with each rule set and combine translation output lattices.

Acknowledgments

This work was supported in part by the GALE program of the Defense Advanced Research Projects

Agency, Contract No. HR0011- 06-C-0022.

References

Graeme Blackwood, Adrià de Gispert, and William Byrne. 2010. Efficient Path Counting Transducers for Minimum Bayes-Risk Decoding of Statistical Machine Translation Lattices. In *Proceedings of ACL, Short Papers*, pages 27–32.

Phil Blunsom, Trevor Cohn, and Miles Osborne. 2008. Bayesian Synchronous Grammar Induction. In *Advances in Neural Information Processing Systems*, volume 21, pages 161–168.

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

Thorsten Brants, Ashok C. Popat, Peng Xu, Franz J. Och, and Jeffrey Dean. 2007. Large language models in machine translation. In *Proceedings of EMNLP-ACL*, pages 858–867.

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

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

Martin Cmejrek, Bowen Zhou, and Bing Xiang. 2009. Enriching SCFG Rules Directly From Efficient Bilingual Chart Parsing. In *Proceedings of IWSLT*, pages 136–143.

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(3).

John DeNero and Dan Klein. 2008. The complexity of phrase alignment problems. In *Proceedings of ACL-HLT, Short Papers*, pages 25–28.

Yonggang Deng and William Byrne. 2008. HMM word and phrase alignment for statistical machine translation. *IEEE Transactions on Audio, Speech, and Language Processing*, 16(3):494–507.

Yonggang Deng. 2005. *Bitext Alignment for Statistical Machine Translation*. Ph.D. thesis, Johns Hopkins University.

Gonzalo Iglesias, Adrià de Gispert, Eduardo R. Banga, and William Byrne. 2009. Rule filtering by pattern for efficient hierarchical translation. In *Proceedings of the EACL*, pages 380–388.

Philipp Koehn, Franz J. Och, and Daniel Marcu. 2003. Statistical phrase-based translation. In *Proceedings of HLT/NAACL*, pages 48–54.

Philipp Koehn. 2010. *Statistical Machine Translation*. Cambridge University Press.

Shankar Kumar, Franz J. Och, and Wolfgang Macherey. 2007. Improving word alignment with bridge languages. In *Proceedings of EMNLP-CoNLL*, pages 42–50.

Yang Liu, Tian Xia, Xinyan Xiao, and Qun Liu. 2009. Weighted alignment matrices for statistical machine translation. In *Proceedings of EMNLP*, pages 1017–1026.

Adam Lopez. 2008. Tera-scale translation models via pattern matching. In *Proceedings of COLING*, pages 505–512.

Daniel Marcu and William Wong. 2002. A phrase-based, joint probability model for statistical machine translation. In *Proceedings of EMNLP*, pages 133–139.

Franz J. Och and Hermann Ney. 2003. A systematic comparison of various statistical alignment models. *Computational Linguistics*, 29(1):19–51.

Franz J. Och. 2003. Minimum error rate training in statistical machine translation. In *Proceedings of ACL*, pages 160–167.

Adam Pauls, Dan Klein, David Chiang, and Kevin Knight. 2010. Unsupervised syntactic alignment with inversion transduction grammars. In *Proceedings of the HLT-NAACL*, pages 118–126.

Markus Saers and Dekai Wu. 2009. Improving phrase-based translation via word alignments from stochastic inversion transduction grammars. In *Proceedings of the HLT-NAACL Workshop on Syntax and Structure in Statistical Translation*, pages 28–36.

Libin Shen, Jinxi Xu, and Ralph Weischedel. 2008. A new string-to-dependency machine translation algorithm with a target dependency language model. In *Proceedings of ACL-HLT*, pages 577–585.

Roy Tromble, Shankar Kumar, Franz J. Och, and Wolfgang Macherey. 2008. Lattice Minimum Bayes-Risk decoding for statistical machine translation. In *Proceedings of EMNLP*, pages 620–629.

Ashish Venugopal, Stephan Vogel, and Alex Waibel. 2003. Effective phrase translation extraction from alignment models. In *Proceedings of ACL*, pages 319–326.

Ashish Venugopal, Andreas Zollmann, Noah A. Smith, and Stephan Vogel. 2008. Wider pipelines: N-best alignments and parses in mt training. In *Proceedings of AMTA*, pages 192–201.

Andreas Zollmann, Ashish Venugopal, Franz J. Och, and Jay Ponte. 2008. A systematic comparison of phrase-based, hierarchical and syntax-augmented statistical MT. In *Proceedings of COLING*, pages 1145–1152.