Long-distance reordering during search for hierarchical phrase-based SMT

Fabienne Braune **Anita Gojun** **Alexander Fraser**
Institute for NLP
Universität Stuttgart
Pfaffenwaldring 5b
D-70569 Stuttgart, Germany
braunefe,gojunaa,fraser@ims.uni-stuttgart.de

Abstract

Long-distance reordering of syntactically divergent language pairs is a critical problem. SMT has had limited success in handling these reorderings during inference, and thus deterministic preprocessing based on reordering parse trees is used. We consider German-to-English translation using Hiero. We show how to effectively model long-distance reorderings during search. Our work is novel in that we look at reordering distances of up to 50 words, and conduct a detailed manual analysis based on a new gold standard.

1 Introduction

Word reordering is a well-known issue in SMT. One successful approach has been to use rule-based preprocessing to reorder parse trees. We would like to perform reordering during inference. Phrase-based hierarchical models (Chiang, 2007) have helped, but reordering over long distances is still a difficult open problem. Consider the following German sentence and English output taken from the hierarchical component of the Moses toolkit (Hoang et al., 2009). These sentences illustrate the successful reordering of the participle *geeinigt (agreed)* from the end of the German clause, to be next to the English auxiliary *have*.

(1) deutschland (germany) , frankreich (france) , israel (israel) und (and) die (the) usa- (us) ***haben (have)*** sich (themselves) im (in) mai (may) 2006 darauf (on) ***geeinigt (agreed)*** , es (it) zu (to) tun (do).

(2) germany , france , israel and the us ***have agreed*** in may 2006 , to do it .

This reordering involves a word movement over 5 tokens and is therefore not a long-distance reordering. However, there can be many more words between the German auxiliary and participle, so the movement required can become arbitrarily large. Restrictions on reordering distance are typically used with hierarchical systems like Hiero because previous experiments have shown some evidence that long-distance reorderings are not effective (Chiang, 2007). We are not aware of careful explorations focusing exclusively on long-distance reorderings in search, prior to our work.

We present the first step towards solving the problem of long-distance reorderings during search. We first analyze the rule geometry required for long-distance German-to-English movement and modify extraction of Hiero's SCFG rules to focus on these rules. We then introduce a new idea, span-width-specific rules in the grammar. By *span*, we denote the number of tokens that are allowed to be covered by a non-terminal symbol (usually "X") in the source language side of an SCFG rule. We define long-distance reordering as occurring over spans containing 11 to 50 source words, and define a new set of rules which apply over spans of 11 to 50 words, which we call *long spans*. We combine these rules (applied on 11 to 50 word spans) with the standard Hiero X rules (applied on 1 to 10 word spans). We further restrict our rules by applying a basic POS-based filtering so that long-span rules contain verbs. Finally, we introduce another innovation to Hiero, which is to block our long-span rules from crossing clause boundaries. We release the source code changes to Hierarchical Moses and our annotated test set for further study by other research groups.

2 Previous Work

The long-distance reordering issue has been considered in phrase-based SMT as well as in syntax-

based SMT. The basic phrase-based model is able to handle word movement up to six tokens but a decrease of performance is observed at higher distortion limits (Koehn et al., 2007). Many reordering methods use a distortion limit between 6 and 9 words (e.g., (Tillmann and Xia, 2003; Koehn et al., 2007; Galley and Manning, 2008)). Green et al. (2010) implement a future cost function and a distortion model that outperform a standard phrase-based system using a distortion limit of 15. We work with longer distances.

Collins et al. (2005) discuss an approach combining rule-based transformations with (phrase-based) SMT. In a preprocessing step, the source language is reordered using parse trees. The restructured output is then provided to a phrase-based MT system. Deterministic preprocessing has several drawbacks such as high sensitivity to parsing errors or the propagation of wrong phrase correspondences (created by incorrect reordering of the training data) into the learned translation probabilities. Preprocessing also does not allow the interaction of long-distance reordering decisions with nearby translation decisions via the language model.

In syntax-based SMT, the size of reordering is given by the span of the grammar rules. In approaches which do not use linguistic syntactic labels (such as ITG (Wu, 1997) or Hiero, where only the start symbol S and the non-terminal X are used), the maximal span size allowed in implementations is often between 10 and 15 tokens, because using wider spans has (in experiments done in the past) resulted in decreased translation quality (e.g., (Chiang, 2007)). Zollmann et al. (2008) expand the span size to 15 only for the translation of short sentences. We present work within the hierarchical phrase-based MT framework that considers rules allowed to span up to 50 words.

Approaches using linguistic syntactic labels (obtained from a source language or target language parser, or both) sometimes also use such span restrictions. However, systems which use source-side-syntactic parses of the test set sometimes do not use such a restriction because they force a match with a syntactic constituent (in the source language parse). There have been many approaches looking at backing off from hard source-side constraints on syntactic labels to Hiero-style X rules (e.g., (Venugopal et al., 2007; Hoang and Koehn, 2010; Mylonakis and Sima'an, 2011)). Due to the diversity of possible structures for German clauses and to poor parse accuracy on long sentences we restrict our study to Hiero, with a view towards integrating soft syntactic constraints (Marton and Resnik, 2008; Chiang, 2010) in the future. Hard syntactic constraints would suffer from too many errors (and too much sparsity) to improve performance in our approach. Our study looks at the specific phenomenon of long-distance reordering in a hierarchical-phrase based framework, by modifying Hiero to support span-width specific rules. We consider exactly the reorderings required for the German-to-English clause reordering problem and focus particular attention on ensuring that the correct reorderings can be considered during search. We employ simple low-knowledge techniques to improve the chances that the correct translation is not only considered but also chosen, but we expect that implementing soft syntactic constraints will improve this further.

The question of handling long-distance movements in hierarchical MT has also been addressed by Sudoh et al. (2010) who present a method that deals with reordering involving connecting together several embedded clauses. Our work differs from (Sudoh et al., 2010) because we handle long-distance reordering inside of a single clause. Moreover, the method by (Sudoh et al., 2010) divides the source language into clauses in a preprocessing step and re-unifies the obtained translations in a post-processing step. In our approach, reordering is performed during inference.

3 Long-Distance Reorderings

In this section, we discuss the type of reordering Hiero is not able to handle, given the constraints used by Chiang (2007). Then we present an analysis of the frequency of such reorderings in a commonly used test set for German-to-English translation. We break this down by the pattern of non-terminals and terminals that will be needed to carry these reorderings out.

Problems with the Hiero Constraints. We first show why hierarchical Moses with standard settings is not able to perform long-distance reordering. To keep the presentation simple, we present example reorderings over distances between 10 and 20 words but our approach handles word movement over 50 words. Consider a German input and its reference translation:[1]

[1]This example is from the WMT 2009 test set, see section 5.

(3) der (the) preis (price) der (of) täglichen (day-to-day) verbrauchsartikel (consumer goods) in den (the) hypermärkten (giant supermarkets) ***ist (is)*** in weniger (less) als (than) 20 monaten (months) um (by) mehr als (over) 30 prozent (percent) ***gestiegen (increased)***.

(4) in the giant supermarkets, the price of day-to-day consumer goods ***soared*** by over 30 percent in less than 20 months.

In order to obtain the reference English translation, the German verbal complex *ist ... gestiegen* has to be translated as a unit into *soared*. The only way to perform this movement using Hiero consists in producing a derivation including a rule of the form:

(5) X $\rightarrow$ < ist (is) X_{10}^{14} X_{15}^{19} gestiegen (increased) ; soared X_2 X_1 >

where the indices on the source non-terminals denote the positions of the source sentence tokens which are covered by X_i^j when the rule is applied, e.g., X_{10}^{14} covers the source language segment *in weniger als 20 monaten*, and the target non-terminals are annotated because they have been swapped (we only annotate the target language side if there is a reordering). The span-width of rule (5) is 12, which corresponds to the sum of the span-widths of the non-terminals and the number of terminals in the rule. In Hiero such a rule cannot be picked during decoding, because only rules with a maximal span of 10 words are allowed. Therefore the translation of the verbal complex *ist ... gestiegen* has to be performed in two steps. Possible rules are:[2]

(6) X $\rightarrow$ < ist (is) X_{10}^{14} ; is X >

(7) X $\rightarrow$ < um mehr (over) X_{17}^{19} gestiegen (increased) ; have increased by more X >

where X_{17}^{19} covers *als 30 prozent*. The complete decoding process yields the malformed English sentence:

(8) the price of daily used in the hypermärkten **is** in less than 20 months **have increased** by more than 30 % .

Besides the movement of the German participle from the end of a German clause to be next to the English auxiliary, other problematic phenomena include the movement of German clause-final particles to be next to the English verb or the reordering of subordinate clauses.

Long-Distance Rule Patterns. We present an analysis of the frequency and shape of sentence pairs in which a correct reordering requires movement over more than 10 tokens. Within the 450 first sentences in the test set of the ACL WMT 2009 German-to-English shared task, we have selected sentence pairs in which the minimal sequence of German tokens on which a hierarchical rule has to be applied to obtain the reference is greater than 10. For instance, in sentence (3), the segment beginning at *ist* and ending at *gestiegen* is the minimal segment in which a reordering has to take place in order to obtain the reference translation (4). The rule for this has to be anchored at the beginning and end of this segment. In other words, its source language side must have the general shape "ist X gestiegen". In the remainder of this paper, we call terminal symbols around a gap *anchor points*. We found 96 sentence pairs in which long-distance reordering is required, which is just over 21% of the sentences we considered. We classify the shapes required into patterns which represent the anchor points as well as the necessary segmentation of the material between those points. Consider the following German sentence and English reference.

Segmentation	Pattern	nb. sent	ratio
One non-term	t^+Xt^+	40	0.42
Two non-terms	t^+XXt^+	23	0.24
More non-terms	$t^+XX^+t^+$	17	0.18
Inversions	X^+t^+,t^+X^+	8	0.08
Others	*No pattern*	8	0.08
Total found sentences		96	1

Figure 1: Patterns of long-distance reordering rules

(9) der (the) ezb (ecb) zufolge (according to) ***wird (will)*** die (the) inflation (inflation) im (in the) jahr (year) 2008 von (from) 2,1 auf (to) 2,5 prozent (percent) ***steigen (rise)***.

(10) according to the ecb , inflation ***will rise*** from 2.1 to 2.5 in the year 2008

A correct reordering of sentence (9) into (10) requires the translation of the segment *die inflation* to move towards the (English) verbal complex while the segment *im jahr 2008 von 2,1 auf 2,5 prozent* has to move behind the complex. Consequently, the source side of a hierarchical rule has to segment these units for reordering them. The pattern of such a rule is the first anchor point (*wird*), two non-terminals covering each reordered segment, followed by the second anchor point (*steigen*). This minimal German pattern can be represented by $tXXt$. We capture rules that involve several terminals around non-terminals by generalizing our patterns (e.g., t^+XXt^+). The patterns for long-distance reorderings in the 450 sentence set are shown in Figure 1.

[2]For such a translation hierarchical Moses can produce a derivation containing more than two rules. To keep the presentation simple, we combine these rules into the two presented.

4 Decoding with Large-Span Rules

We have shown that hierarchical rules for long-distance reordering have a particular shape on source language side. Basing on this observation, we modify the grammar and decoding procedure of hierarchical Moses to build a system which can capture the specificity of such reorderings.

Creating special rules for long-distance reordering. In a first step, we extract rules designed for long-distance reordering, that is rules that have a more specific geometry than standard hierarchical rules. By "specific geometry", we denote rules that match the patterns presented in section 3. We want these rules only to be considered when long-distance reordering is required. In order to achieve this, we define different spans on which our rules are allowed to be used during decoding. In other words, we build a Hiero grammar consisting of two subsets which apply on different spans during decoding. The first set contains all Hiero rules extracted using the standard procedure. Rules belonging to this set apply to spans having size from 1 to 10. The second set contains rules with the following properties:

(i) Instead of having one aligned terminal on each side of a rule, we require each source side non-terminal except the first to have at least one aligned terminal on its left and one on its right.

(ii) In each rule extracted following constraint (i) we allow non-terminal symbols to be further split into adjacent non-terminals.

Rules extracted following constraints (i) and (ii) build an SCFG grammar with rules having the same shape, on the source language side, as the patterns presented in section 3. Note that because we allow the first non-terminal of each rule to have no terminal on its left, we also capture patterns of the form X^+t^+ but not t^+X^+. Because these rules are specifically designed for long-distance reordering they are only used on spans having size between 11 and 50 in decoding.

The creation of a specific SCFG grammar for large spans allows the handling of long-distance reordering while keeping the set of hierarchical rules acceptably small. Our set of rules for long-distance reordering are extracted on spans from 1 to 10. The decision to extract on small spans is based on the observation that most rules needed for the long-distance reorderings required to reorder German clauses can be found in short span examples. We found that rules extracted from longer spans were noisy and rarely correct and that the rules for many examples of long-distance reorderings which are present in the training data can not be extracted because noisy alignments incorrectly block extraction. Rules are scored by computing maximum likelihood estimation using phrase counts as described in (Chiang, 2007).

Let us illustrate the functioning of large-span rules. Consider again the sentence pair presented in section 3. In a system containing large-span rules, the rule $\rightarrow$ <ist (is) X X gestiegen (increased) ; soared X_2 X_1> is extracted in training and applied on spans between 11 and 50 at decoding. Hence, the rule necessary for a correct translation of our example sentence is available in our extended system.

Decoding with long-distance rules. The hierarchical Moses decoder allows the user to work with multiple sets of hierarchical rules having different maximal span sizes. However, the possibility to decode using rules with span greater than a minimal threshold was not implemented in hierarchical Moses.[3] In order to overcome this problem, we have defined a new type of grammar for which the lookup procedure only selects rules greater than a given minimal span.

Making long-distance rules reachable. Creating a set of rules applying on spans 11 to 50 during decoding is not sufficient to allow our modified system to effectively use large-span rules. In order to be applied, a hierarchical rule must be reachable, meaning that there must be a valid derivation for the subtree covered by the non-terminal X_i^j in the source language side of the considered rule. Because hierarchical rules can only apply on spans up to 10, those rules can only cover sub-spans of X_i^j when $j - i$ is smaller than 11. Even when allowing adjacent non-terminals, this size is likely to be greater than 10. If no other grammar is accessible to the decoder, these partial translations cannot be combined sequentially and for spans greater than 10, large-span rules have to be applied recursively. This massively restricts the applicability of long-distance rules. It is important to note here that in hierarchical Moses glue rules can only be applied

[3] This is due to the fact that rule-lookup is done in an incremental fashion. For each type of grammar provided to the decoder, the lookup procedure only selects rules for which all subspans have already been explored.

on partial translations of an entire sentence.[4] In other words, a glue rule has the form S → < S X ; S X > , where S corresponds to the beginning of a sentence whereas a rule for sequentially combining segments under a considered span should have the shape X → < $XX; XX$ > . To make our large-span rules reachable, we augment our system with this rule. In summary, our decoder has access to four different grammar rule tables: (i) the two standard "S" rules (ii) the full set of Hiero rules on spans smaller than 10 (iii) rules with specific geometry on spans of size 11 to 50 (iv) an X → < $XX; XX$ > rule on spans of size 1 to 50.

Filtering out poorly informative rules. Even when performing a rule extraction procedure with a constraint on non-terminals the following rules are part of the extracted grammar.

(11) X → < der (the) X , ; the X , >

(12) X → < , X . ; , X . >

Such rules are not useful for achieving long-distance reordering. Moreover, they tend to get high translation scores and are likely to be chosen often during decoding. This factor contributes to the fact that after tuning with MERT, the weight assigned to the count feature of large-span rules is too low to allow the required reordering to take place. We address this problem by using a very simple filter on the grammar operating on large spans. We only keep rules that contain at least one verb on source and target language side.[5]

Clausal boundary restriction. The hierarchical patterns for long-distance reordering rules identified in section 3 are intra-clausal patterns. This means that they only apply inside of a single clause, which can, however, contain embedded clauses. In other words, when a pattern of the form $t^+X^+t^+$ in the source language side of the rule is matched to a segment which begins in one clause and ends in another clause, then the rule is likely to be wrongly anchored. As an illustration, consider source language sentence (13) and rule (14) where the non-terminal X covers token 3 to 12.

(13) er (he) **ist (is)** als (as) solist (solist) unterwegs (travelling) und (and) **hat (has)** seine (his) karriere (career) eher (rather) im westen (in the west) **aufgebaut (built)** .

(14) X → < ist (is) X_3^{12} aufgebaut (built) ; is built X >

The anchor points *ist* and *aufgebaut* match two verbs that do not belong to the same complex. This erroneously reorders the participle *built* next to the verb *is* (instead of *has*). Consequently, a malformed sentence like (15) is generated.

(15) he **is built** as solist traveling and **has** his career more in the west

This can be avoided by forcing rules to be applied inside of a single clause. This is achieved by extracting clause boundaries from the parse tree of each source language sentence.[6] Clauses are then represented as intervals delimited by the identified boundaries. The constraint we enforce regarding clause boundaries works as follows: if the first terminal of a rule is inside of a clause, then the last terminal of the same rule has to be inside of the same clause. In our example, if the starting point of a rule is at a position between 0 and 5, then its end position has to be smaller or equal to 6. This restriction allows the avoidance of all wrong anchoring related to the crossing of clause boundaries. For instance, in sentence (13) above, rule (14) would not be allowed to apply because its first terminal is at position 1 in the source sentence while its second terminal is at position 12. Note that this also handles embedded clauses correctly.

5 Experimental Setup

The baseline system for our experiments is hierarchical Moses with a span size up to 50 tokens instead of 10 in the standard settings. Enabling hierarchical Moses to reorder over long distances involves two main modifications. First, hierarchical rules have to be extracted for spans having a maximal size of 50 tokens instead of 10. Second, the decoder has to be allowed to pick rules with span size 50. Extraction of hierarchical rules on spans containing up to 50 tokens is intractable in terms of cpu time and disk space. In order to nevertheless work with such a system we adopt the same strategy as described in section 4: we extract rules on spans up to 10 and allow the obtained grammar to apply to spans up to 50 words during decoding. The modified system presented in section 4 will be evaluated against this baseline. Note that choosing a baseline with extended span size allows us to evaluate our approach against a system enabled to perform long-distance reordering. The results obtained by hierarchical Moses with standard settings

[4] This corresponds to Chiang's definition of glue-rules in (Chiang, 2007).

[5] We tag the German and English parallel training corpus with TreeTagger, and discard extracted rule tokens which do not contain a verb on both sides; we then delete the POS tags.

[6] We use BitPar (Schmid, 2004) to extract clause boundaries. Boundaries correspond to the position of the token labeled by the opening and closing S-Nodes in the parse tree.

on all test sets is also provided, but since it can not perform long-distance reorderings we provide no further analysis.

The translation model has been trained using 1,502,301 bilingual sentences after length ratio filtering. GIZA++ (Och and Ney, 2003) has been used for generating the word alignments, combined with the grow-diag-final-and heuristic (Koehn et al., 2007). We trained our monolingual 5-gram language model using the English side of the training data. Feature weights are tuned using Pairwise-Ranked optimization (Hopkins and May, 2011) followed by standard MERT line search (for fine tuning of the length penalty). We evaluate two tasks. For the ACL WMT 2009 German-to-English shared task, we use news-dev2009a as our dev set, and news-dev2009b as our test set. To reduce the effect of data sparsity for the difficult task of long-distance reordering, we also consider a Europarl translation task, using the same system (with the same training data), but using Europarl test2007 as our dev set, and Europarl dev2006 as our test set.

6 Evaluation

We perform a two step evaluation procedure. First the compared systems are evaluated using automatic metrics. In a second step we compare the systems using a manually annotated test set.

Automatic Evaluation. As a first automatic evaluation metric, we use 4-gram BLEU (Papineni et al., 2002). Because BLEU does not consider the positions of matched n-grams and does not capture the distance of erroneous reorderings, we use LRscore (Birch and Osborne, 2011) as a second metric to evaluate reordering quality. This method compares the alignments between input and reference with the alignments between input and system output (Kendall's Tau over permutations is used as the distance metric). We provide two measures (i) LRscore as proposed in (Birch and Osborne, 2011) where the interpolation parameter[7] α is set to 0.2623 (ii) reordering performance only, i.e., $\alpha = 1$.

Figure 2 shows the results for all systems on the Europarl and ACL WMT 2009 tasks. Our improved hierarchical system is denoted by Improved-50. Hierarchical Moses with span sizes up to 50 tokens is Std-50. Hierarchical Moses with standard settings is denoted by Std. On the Europarl translation task, Std-50 and Improved-50 achieve a similar performance in terms of BLEU while Improved-50 obtains a 0.31 better LRscore when considering the reordering distance only ($\alpha = 1$). When using interpolated LRscore, Improved-50 is 0.26 better than Std-50. On a test set belonging to the same genre as the training set, improved-50 provides better reordering quality. On ACL WMT 2009, Improved-50 obtains a 0.4 worse BLEU score than Std-50 together with a 0.4 improvement in LRscore when considering the reordering distance only. The interpolated LRscore metric shows a 0.2 improvement of Improved-50 over Std-50. On a test set belonging to a genre different than the training set, Improved-50 causes a small decrease in BLEU together with somewhat better reordering. The decrease in BLEU observed is mainly bad lexical choice caused by using rules on a different domain.

Manual Evaluation. In a second step, we report the amount of correct and incorrect long-distance reordering performed by the evaluated systems on a manually annotated test set. Our test set consists of the 450 sentences presented in section 3. For counting correct reordering, we consider each sentence in our set and evaluate the translation of its source language pattern. We look at the anchor points t as well as the segments represented by X. We provide two types of counts (i) **reference matches** and (ii) **human matches**. A **reference match** requires the translation of the anchor points t to be in the same order and have the same surface form as in the reference translation. We also require the segments covered by X to be in the same order as in the reference translation. A **human match** includes translations where the reordering of the anchor points t is the same as in the reference, but we don't require the translation of t to have the same surface form as in the reference. We also allow the ordering of the segments covered by X to be different than in the reference as long as it is considered as correct by our human annotator. As an illustration for the difference between reference and human matches, consider again source sentence (3) and reference (4). Also consider the following possible translation of (3):

(16) the price of day-to-day consumer goods in supermarkets increased in less than 20 months by over 30 percent.

Sentence (16) cannot be considered as a **reference match** because it translates *ist ... gestiegen* into *increased* instead of *soared* and because the seg-

[7] This parameter controls the trade-off with BLEU.

System	BLEU (dev)	BLEU (test)	LRscore ($\alpha = 1$)	LRscore ($\alpha = 0.2326$)
Improved-50 (Europarl)	29.24	28.32	70.38	60.60
Std-50 (Europarl)	29.49	28.27	70.07	60.34
Std (Europarl)	29.13	28.00	70.82	60.68
Improved-50 (ACL WMT 2009)	18.86	18.91	67.92	56.52
Std-50 (ACL WMT 2009)	18.77	19.30	67.52	56.33
Std (ACL WMT 2009)	18.54	19.30	67.54	56.32

Figure 2: Europarl and ACL WMT 2009 German-to-English shared tasks

ments *in less than 20 months* and *by over 30 percent* are reversed. This sentence is, however, **a human match**. Each reference match is also a human match and all human matches are counted as **correct**. We make the simplifying assumption that each reordering involving a large-span rule on a sentence which is not in our set is **wrong**. We provide a further count denoted by **pattern match** which includes all cases where the source side pattern has been matched using a large-span rule but where the system nevertheless yielded an incorrect translation. As will be shown below this measure allows us to evaluate the potential of a grammar to apply long-distance reordering rules even when the translation is wrong. We also report cases where a system is able to reorder over distances greater than 10 words by gluing together rules that translate the edges of the reordering. A correct translation of sentence 19 can be obtained, for instance, by using rules 17 and 18:

(17) X $\rightarrow <$ wird (will) X_5^6 ; X will $>$

(18) X $\rightarrow <$ X_7^8 2008 X_{10}^{14} steigen (increase) ; increase X X 2008 $>$

(19) according to the ecb , inflation ***will rise*** from 2.1 to 2.5 in the year 2008

Note that this strategy only allows the performance of a restricted amount of long-distance reorderings: sentences similar to 3 cannot be reordered in this way, and word movements cannot be over a distance of more than 22 words.

Figure 3 shows the amount of correct and incorrect long-distance reordering performed by Std-50 and Improved-50 on our manually annotated test set.[8] For Std-50 we observed 14 cases where long-distance reordering is performed where not required (on sentences outside of our selected sentences). Std-50 correctly reorders 9 sentences with the gluing strategy described above. Std-50 is able to correctly match a source side pattern in only 17 cases. When a pattern has been matched, the system is generally able to correctly translate it. The 17 pattern matches of Std-50 yield 13 correct translations. Reference matches are very rare. This is mainly due to the fact that the translation of the anchor points t in the source side of a rule have a different surface form than in the reference. The accuracy of Std-50 in applying large-span rules on sentences where long-distance reordering has to be performed is poor: the amount (14) of reorderings performed on wrong sentences is approximately the same as the amount (17) of German pattern matches. This last observation also explains the poor reordering quality observed on Europarl. Improved-50 matches twice as many source language patterns as Std-50 while performing half as many reorderings on wrong sentences (Figure 3). Improved-50 does a better job in identifying the correct context for application for large-span rules. This system also performs correct long-distance reordering in 24 cases compared to only 14 for Std-50. Again, this represents an improvement over Std-50, but the amount of pattern matches still represents only 35.4% of our test set. Further study is required to determine if this is primarily due to having no rules that could match, or instead because monotonic derivations have a better score. Finally, out of 34 correct pattern matches, 24 yield a correct translation, so the translation is correct in 70% of the matches. We plan to improve the ability of our system to provide a correct translation when a correct source language pattern match is made. We observed 7 cases where long-distance reordering is erroneously performed on sentences outside of our annotated set. The system correctly reorders 8 sentences with the gluing strategy described above. Overall, Improved-50 outperformed Std-50, indicating we have made progress on the difficult problem of long-distance reordering, but there is more work to be done.

7 Conclusion

Long-distance reorderings are required in about 21% of the German sentences in news-test2009b. Simply dropping the span restriction of hierarchi-

[8] Because Std cannot perform any long-distance reordering (because of its span restriction), it has no matches.

	Std-50			Improved-50		
Pattern	Reference	Human	Pattern Match	Reference	Human	Pattern Match
t^+Xt^+	2	9	10	2	11	19
t^+XXt^+	0	1	2	0	2	5
$t^+XX^+t^+$	0	1	1	0	6	6
X^+t^+ or t^+X^+	1	1	2	1	1	1
No general pattern	1	2	1	1	2	1
Total	4	14	17	4	23	36

Figure 3: Evaluation of the reorderings in our 450 sentence set, broken down by pattern type. Std-50 performs 14 reorderings on sentences where no reordering is necessary; Improved-50 performs 7.

cal Moses results in poor long-distance reordering. We presented an improved version of hierarchical Moses including (i) a specific set of rules for long-distance reordering made reachable and adequately filtered (ii) a decoding procedure using different span-widths (iii) clausal boundary restrictions. Our improved system performs more long-distance reorderings, accurately selects the context of application of large-span rules, and also correctly translates in many cases.

Acknowledgments

This work was funded by Deutsche Forschungsgemeinschaft grant Models of Morphosyntax for Statistical Machine Translation. This work was supported in part by the IST Programme of the European Community, under the PASCAL2 Network of Excellence, IST-2007-216886.

References

Birch, Alexandra and Miles Osborne. 2011. Reordering metrics for mt. In *ACL*.

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

Chiang, David. 2010. Learning to translate with source and target syntax. In *ACL*.

Collins, Michael, Philipp Koehn, and Ivona Kucerova. 2005. Clause restructuring for statistical machine translation. In *ACL*.

Galley, Michel and Christopher D. Manning. 2008. A simple and effective hierarchical phrase reordering model. In *EMNLP*.

Green, Spence, Michel Galley, and Christopher D. Manning. 2010. Improved models of distortion cost for statistical machine translation. In *NAACL-HLT*.

Hoang, Hieu and Philipp Koehn. 2010. Improved translation with source syntax labels. In *ACL WMT*.

Hoang, Hieu, Philipp Koehn, and Adam Lopez. 2009. A unified framework for phrase-based, hierarchical, and syntax-based SMT. In *IWSLT*.

Hopkins, Mark and Jonathan May. 2011. Tuning as ranking. In *EMNLP*.

Koehn, Philipp, H Hoang, A Birch, C Callison-Burch, M Federico, N Bertoldi, B Cowan, W Shen, C Moran, R Zens, C Dyer, O Bojar, A Constantin, and E Herbst. 2007. Moses: Open source toolkit for statistical machine translation. In *ACL*.

Marton, Yuval and Philip Resnik. 2008. Soft syntactic constraints for hierarchical phrased-based translation. In *ACL-HLT*.

Mylonakis, Markos and Khalil Sima'an. 2011. Learning hierarchical translation structure with linguistic annotations. In *ACL-HLT*.

Och, Franz Josef and Hermann Ney. 2003. A systematic comparison of various statistical alignment models. *Computational Linguistics*, 29(1).

Papineni, Kishore, Salim Roukos, Todd Ward, and Wei jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *ACL*.

Schmid, Helmut. 2004. Efficient parsing of highly ambiguous context-free grammars with bit vectors. In *COLING*.

Sudoh, Katsuhito, Kevin Duh, Hajime Tsukada, Tsutomu Hirao, and Masaaki Nagata. 2010. Divide and translate: Improving long distance reordering in statistical machine translation. In *ACL WMT*.

Tillmann, Christoph and Fei Xia. 2003. A phrase-based unigram model for statistical machine translation. In *NAACL-HLT*.

Venugopal, Ashish, Andreas Zollmann, and Stephan Vogel. 2007. An efficient two-pass approach to synchronous-CFG driven statistical MT. In *NAACL*.

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

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