Efficient solutions for word reordering in German-English phrase-based statistical machine translation

Arianna Bisazza and **Marcello Federico**
Fondazione Bruno Kessler
Trento, Italy
{bisazza,federico}@fbk.eu

Abstract

Despite being closely related languages, German and English are characterized by important word order differences. Long-range reordering of verbs, in particular, represents a real challenge for state-of-the-art SMT systems and is one of the main reasons why translation quality is often so poor in this language pair. In this work, we review several solutions to improve the accuracy of German-English word reordering while preserving the efficiency of phrase-based decoding. Among these, we consider a novel technique to dynamically shape the reordering search space and effectively capture long-range reordering phenomena. Through an extensive evaluation including diverse translation quality metrics, we show that these solutions can significantly narrow the gap between phrase-based and hierarchical SMT.

1 Introduction

Modeling the German-English language pair is known to be a challenging task for state-of-the-art statistical machine translation (SMT) methods. A major factor of difficulty is given by word order differences that yield important long-range reordering phenomena.

Thanks to specific reordering modeling components, phrase-based SMT (PSMT) systems (Zens et al., 2002; Koehn et al., 2003; Och and Ney, 2002) are generally good at handling *local* reordering phenomena that are not captured inside phrases. However, they typically fail to predict long reorderings. On the other hand, hierarchical SMT (HSMT) systems (Chiang, 2005) can learn reordering patterns by means of discontinuous translation rules, and are therefore considered a better choice for language pairs characterized by massive and hierarchical reordering.

Looking at the results of the Workshop of Machine Translation's last edition (WMT12) (Callison-Burch et al., 2012), no particular SMT approach appears to be clearly dominating. In both language directions (official results excluding the online systems) the rule-based systems outperformed all SMT approaches, and among the best SMT systems we find a variety of approaches: pure phrase-based, phrase-based and hierarchical systems combination, n-gram based, a rich syntax-based approach, and a phrase-based system coupled with POS-based pre-ordering. This gives an idea of how challenging this language pair is for SMT and raises the question of which SMT approach is best suited to model it.

In this work, we aim at answering this question by focussing on the word reordering problem, which is known to be an important factor of SMT performance (Birch et al., 2008). We hypothesize that PSMT can be as successful for German-English as the more computationally costly HSMT approach, provided that the reordering-related parameters are carefully chosen and the best available reordering models are used. More specifically, our study covers the following topics: distortion functions and limits, and dynamic shaping of the reordering search space based on a discriminative reordering model.

We first review these topics, and then evaluate them systematically on the WMT task using both generic and reordering-specific metrics, with the aim of providing a reference for future system developers' choices.

2 Background

Word order differences between German and English are mainly found at the clause (global) level, as opposed to the phrase (local) level. We refer to Collins et al. (2005) and Gojun and Fraser (2012) for a detailed description of the German clause structure. To briefly summarize, we can say that

Proceedings of the Eighth Workshop on Statistical Machine Translation, pages 440–451,
Sofia, Bulgaria, August 8-9, 2013

the *verb-second order* of German main clauses contrasts with the rigid SVO structure of English, as does the clause-final verb position of German subordinate clauses. A further difficulty is given by the German *discontinuous* verb phrases, where the main verb is separated from the inflected auxiliary or modal. The distance between the two parts of a verb phrase can be arbitrarily long as shown in the following example:

[DE] Jedoch **konnten** sie Kinder in Teilen von Helmand und Kandahar im Süden aus Sicherheitsgrund **nicht erreichen**.

[EN] But they **could not reach** children in parts of Helmand and Kandahar in the south for security reasons.

Translating this sentence with a PSMT engine implies performing two very long jumps that are not even considered by typical systems employing a distortion limit of 6 or 8 words. At the same time, increasing the distortion limit to very high values is known to have a negative impact on both efficiency and translation quality (cf. results presented later in this paper).

Because reordering patterns of this kind are very common between German and English, this paper focuses on techniques that enable the PSMT decoder to explore long jumps and thus improve reordering accuracy without hurting efficiency nor general translation quality.

2.1 Alternative approaches

German-English reordering in SMT has been widely studied and is still an open topic. In this work, we only consider efficient solutions that are fully integrated into the decoding process, and that do not require syntactic parsers or manual reordering rules. Still, it has to be mentioned that several alternative solutions were proposed in the literature. A well-known strategy consists of pre-ordering the German sentence in an English-like order by applying a set of manually written rules to its syntactic parse tree (Collins et al., 2005).[1] Other approaches learn the pre-ordering rules automatically, from syntactic parses (Xia and McCord, 2004; Genzel, 2010) or from part-of-speech labels (Niehues and Kolss, 2009). In the former case, pre-ordering decisions are typically taken deterministically (i. e. one permuation per sentence), whereas in the latter, multiple alternatives are represented as word lattices, and the optimal path is selected by the decoder at translation time. In (Tromble and Eisner, 2009), pre-ordering is cast as a permutation problem and solved by a model that estimates the probability of reversing the relative order of any two input words.

In the field of tree-based SMT, positive results in German-English were achieved by combining syntactic translation rules with unlabeled hierarchical SMT rules (Hoang and Koehn, 2010). More recently, Braune et al. (2012) proposed to improve the long-range reordering capability of an HSMT system by integrating constraints based on clausal boundaries and by manually selecting the rule patterns applicable to long word spans. The paper did not analyse the impact of the technique on efficiency.

2.2 Evaluation methods

A large number of previous works on word reordering measured their success with general-purpose metrics such as BLEU (Papineni et al., 2001) or METEOR (Banerjee and Lavie, 2005). These metrics, however, are only indirectly sensitive to word order and do not sufficiently penalize long-range reordering errors, as demonstrated for instance by Birch et al. (2010). While BLEU remains a standard choice for many evaluation campaigns, we believe it is extremely important to complement it with metrics that are specifically designed to capture word order differences. In this work, we adopt two reordering-specific metrics in addition to BLEU and METEOR:

Kendall Reordering Score (KRS). As proposed by Birch et al. (2010), the KRS measures the similarity between the input-output reordering and the input-reference reordering. This is done by converting word alignments to permutations and computing a permutation distance among them. When interpolated with BLEU, this score is called LRscore.[2]

Verb-specific KRS (KRS-V). The ideal way to automatically evaluate our systems would be to use syntax- or semantics-based metrics, as the impact of long reordering errors is particularly important at these levels. As a light-weight alternative, we instead concentrate the evaluation on those word classes that are typically crucial to guess the general structure of a sentence. To this end, we adopt a word-weighted version of the

[1] A similar solution for the opposite translation direction (English-German) was proposed by Gojun and Fraser (2012).

[2] Thus, our KRS results correspond exactly to the LRscore(α=1) presented in other papers.

KRS and set the weights to 1 for verbs and 0 for all other words, so that only verb reordering errors are captured. We call the resulting metric KRS-V. The KRS-V rates a translation hypothesis as perfect (100%) when the translations of all source verbs are located in their correct position, regardless of the other words' ordering.

3 Early distortion cost

In its original formulation, the PSMT approach includes a basic reordering model, called **distortion cost**, that exponentially penalizes longer jumps among consecutively translated phrases simply based on their distance. Thus, a completely monotonic translation has a total distortion cost of zero.

A weakness of this model is that it penalizes long jumps only when they are performed, rather than accumulating their cost gradually. As an effect, hypotheses with gaps (i. e. uncovered input positions) can proliferate and cause the pruning of more monotonic hypotheses that could lead to overall better translations.

To solve this problem, Moore and Quirk (2007) proposed an improved version of the distortion cost function that anticipates the gradual accumulation of the total distortion cost, making hypotheses with the same number of covered words more comparable with one another. **Early distortion cost** (as called in Moses, or "distortion penalty estimation" in the original paper) is computed by a simple algorithm that keeps track of the uncovered input positions. Note that this option affects the distortion *feature function*, but not the distortion *limit*, which always corresponds to the maximum distance allowed between consecutively translated phrases.

Early distortion cost was shown by its authors to yield similar BLEU scores as the standard one but with stricter pruning parameters, i. e. faster decoding. Experiments were performed on an English-French task, with a fixed distortion limit of 5 and without lexicalized reordering models. Our study deals with a language pair that is arguably more difficult at the level of reordering. Moreover, we start from a stronger baseline and measure the impact of early distortion cost in various distortion limit settings, using also reordering-specific metrics. Results are presented in Section 6.2.

4 Word-after-word reordering modeling and pruning

Phrase orientation (lexicalized reordering) models (Tillmann, 2004; Koehn et al., 2005; Galley and Manning, 2008) have proven very useful for short and medium-range reordering and are probably the most widely used in PSMT nowadays. However, their coarse classification of reordering steps makes them unsuitable to capture long-range reordering phenomena, such as those attested in German-English. Indeed, Galley and Manning (2008) reported a decrease of translation quality when the distortion limit was set beyond 6 in Chinese-English and beyond 4 in Arabic-English.

To address this problem, we have developed a different reordering model that predicts what input word should be translated at a given decoding state (Bisazza, 2013; Bisazza and Federico, 2013). The model is similar to the one proposed by Visweswariah et al. (2011), however we use it differently: that is, not simply for data preprocessing but as an additional feature function fully integrated in the phrase-based decoder. More importantly, we propose to use the same model to dynamically shape the space of reorderings explored during decoding (cf. Section 4.2), which was never done before.

Another related work is the source-side decoding sequence model by Feng et al. (2010), that is a generative n-gram model trained on a corpus of pre-ordered source sentences. Although reminiscent of a source-side bigram model, our model has two important differences: (i) the discriminative modeling framework enables us to design a much richer feature set including, for instance, the context of the next word to pick; (ii) all our features are independent from the decoding history, which allows for an efficient decoder-integration with no effect on hypothesis recombination.

Finally, we have to mention the models by Al-Onaizan and Papineni (2006) and Green et al. (2010), who predict the direction and (binned) length of a jump to perform after a given input word. Those models too were only used as additional feature functions, and were not shown to maintain translation quality and efficiency at very high distortion limits.

4.1 The model

The Word-after-word (**WaW**) reordering model is trained to predict whether a given input position

should be translated *right after* another, given the words at those positions and their contexts. It is based on the following maximum-entropy binary classifier:

$$P(R_{i,j}{=}Y|f_1^J, i, j) = \frac{exp[\sum_m \lambda_m h_m(f_1^J, i, j, R_{i,j}{=}Y)]}{\sum_{Y'} exp[\sum_m \lambda_m h_m(f_1^J, i, j, R_{i,j}{=}Y')]}$$

where f_1^J is a source sentence of J words, h_m are feature functions and λ_m the corresponding feature weights. The outcome Y can be either 1 or 0, with $R_{i,j}{=}1$ meaning that the word at position j is translated right after the word at position i.

Training examples are extracted from a corpus of reference reorderings, obtained by converting the word-aligned parallel data into a set of source sentence permutations. A heuristic similar to the one proposed by Visweswariah et al. (2011) is used to this end. For each input word, we generate: (i) one positive example for the word that should be translated right after it; (ii) negative examples for all the uncovered words that lie within a given *sampling window* or δ. The latter parameter serves to control the proportion between positive and negative examples.

The WaW model builds on binary features that are extracted from the local context of positions i and j, and from the words occurring between them. In addition to the actual words, the features may include POS tags and shallow syntax labels (i. e. chunk types and boundaries). For instance, one feature may indicate that the last translated word (w_i) is an adjective while the currently translated one (w_j) is a noun:

$$\text{POS}(w_i)\text{=adj} \wedge \text{POS}(w_j)\text{=noun}$$

Other features indicate that a given word or punctuation is found between w_i and w_j:

$$w_b\text{='jedoch'} \quad ... \quad w_b\text{='.'}$$

or that w_i and w_j belong to the same shallow syntax chunk.

The WaW reordering model can be seamlessy integrated into a standard phrase-based decoder that already includes phrase orientation models. When a partial hypothesis is expanded with a given phrase pair, the model returns the log-probability of translating its words in the order defined by the phrase-internal word alignment. Moreover, the global WaW score is independent from phrase segmentation, and normalized across outputs of different lengths.

The complete list of features, training data generation algorithm and other implementation details are presented in (Bisazza, 2013) and (Bisazza and Federico, 2013).

4.2 Early reordering pruning

Besides providing an additional feature function for the log-linear PSMT framework, the WaW model's predictions can be used as an early indication of whether or not a given reordering path should be further explored. In fact, we have mentioned that the existing reordering models are not capable of guiding the search through very large reordering search spaces. As a solution, we propose to decode with loose reordering constraints (i. e. high distortion limit) but only explore those long reorderings that are promising according to the WaW model.

More specifically, at each hypothesis expansion, we consider the set of input positions that are reachable within the fixed distortion limit. Only based on the WaW score, we apply histogram and threshold pruning to this set and then proceed to expand only the non-pruned positions.[3] Furthermore, it is possible to ensure that local reorderings are always allowed, by setting a so-called *non-prunable-zone* of width ϑ around the last covered input position.[4] In this way, we can ensure that the usual space of short to medium-range reordering is exhaustively explored in addition to few promising long-range reorderings.

The rationale of this approach is two-fold: First, to avoid costly hypothesis expansions for very unlikely reordering steps and thus speed up decoding under loose reordering constraints. Second, to decrease the risk of model errors by exploiting the fact that some components of the PSMT log-linear model are more important than others at different stages of the translation process.

The WaW model is not the only scoring function that can be used for early reordering pruning. In principle, even phrase orientation model scores could be used, but we expect them to perform poorly due to the coarse classification of reordering steps (all phrases that are not adjacent to the current one are treated as *discontinuous* steps).

[3] The idea is reminiscent of early pruning by Moore and Quirk (2007): an optimization technique that consists of discarding hypothesis extensions based on their estimated score *before* computing the exact language model score.

[4] See (Bisazza, 2013) for technical details on the integration of word-level pruning with phrase-level hypothesis expansion.

5 Reordering in hierarchical SMT

To allow for a fair evaluation of our systems, we also perform a contrastive experiment using a tree-based SMT approach: namely, **hierarchical phrase-based SMT (HSMT)** (Chiang, 2005).

Reordering in HSMT is not modeled separately but is embedded in the translation model itself, which contains lexicalized, non syntactically motivated rules that are directly learnt from word-aligned parallel text. The major strength of HSMT compared to PSMT, is the ability to learn discontinous phrases and long-range lexicalized reordering rules. However, this modeling power has a cost in terms of model size and decoding complexity.

To have a concrete idea, consider that the phrase-table trained on our SMT training data (cf. Section 6.1) with a maximum phrase length of 7 contains 127 million entries (before phrase table pruning). The hierarchical rule table trained on the same data with a comparable span constraint (10) contains instead 1.2 billion entries – one order of magnitude larger.

Furthermore, the HSMT decoder is based on a chart parsing algorithm, whose complexity is cubic in the input length, and even higher when taking into account the target language model. This issue can be partially addressed by different strategies such as cube pruning (Chiang, 2007), which reduces the LM complexity to a constant, or rule application constraints. One of such constraints is the maximum number of source words that may be covered by non-terminal symbols (span constraint). Setting a span constraint – which is essential to obtain reasonable decoding times – means preventing long-range reordering similarly to setting a distortion limit in PSMT. In our experiments, we consider two settings for this parameter: 10 to capture short to medium-range reorderings, and 20 to also capture long-range reorderings.

6 Experiments

In this section we evaluate the impact on translation quality and efficiency of the techniques presented above. Our main objective is to empirically verify the hypothesis that better reordering modeling and better reordering space definition can significantly improve the accuracy of PSMT in German-English without sacrificing its efficiency.

6.1 Experimental setup

We choose the WMT German-English news translation task as our case study. More specifically we use the WMT10 training data: Europarl (v.5) plus News-commentary-2010 for a total of 1.6M parallel sentences, 44M German tokens. The target LM is trained on the monolingual news data provided for the constrained track of WMT10 (1133M English tokens). For development we use the WMT08 news benchmark, while for testing we use the following data sets:

tests(09-11): the concatentation of three previous years' benchmarks from 2009 to 2011 (8017 sentences, 21K German tokens).

test12: the latest released benchmark (3003 sentences, 8K German tokens).

Each data set includes one reference translation. Note that our goal is not to reach the performance of the best systems participating at the last WMT edition, but rather to assess the usefulness of our techniques on a larger and therefore more reliable test set, while starting from a reasonable baseline.[5]

For German tokenization and compound splitting we use Tree Tagger (Schmid, 1994) and the Gertwol morphological analyser (Koskenniemi and Haapalainen, 1994).[6]

All our SMT systems are built with the Moses toolkit (Koehn et al., 2007; Hoang et al., 2009), and word alignments are generated by the Berkeley Aligner (Liang et al., 2006). The target language model is estimated by the IRSTLM toolkit (Federico et al., 2008) with modified Kneser-Ney smoothing (Chen and Goodman, 1999).

The **phrase-based** baseline decoder includes a phrase translation model (two phrasal and two lexical probability features), a lexicalized reordering model (six features), a 6-gram target language model, distortion cost, word and phrase penalties. As lexicalized reordering model, we use a hierarchical phrase orientation model (Galley and Manning, 2008) trained on all the parallel data using three orientation classes – *monotone, swap* or *discontinuous* – in bidirectional mode. Statistically

[5] Our results on test12 are not directly comparable to the WMT12 submissions due to the different training data: that is, the WMT12 parallel data includes 50M German tokens of Europarl data and 4M of news-commentary, as opposed to the 41M and 2.5M released for WMT10 and used in our experiments.

[6] http://www2.lingsoft.fi/cgi-bin/gertwol

improbable phrase pairs are pruned from the translation model as proposed by Johnson et al. (2007).

The **hierarchical** system is trained and tested using the standard Moses configuration which includes: a rule table (two phrasal and two lexical probability features), a 6-gram target language model, word and rule penalties. We set the span constraint (cf. Section 5) to the default value of 10 words for rule extraction, while for decoding we consider two different settings: the default 10 words and a large value of 20 to enable very long-range reorderings.

Feature weights for all systems are optimized by minimum BLEU-error training (Och, 2003) on test08. To reduce the effects of the optimizer instability, we tune each configuration four times and use the average of the resulting weight vectors for testing, as suggested by Cettolo et al. (2011).

The source-to-reference word alignments that are needed to compute the reordering scores are generated by the Berkeley Aligner previously trained on the training data. Source-to-output alignments are obtained from the decoder's trace.

6.2 Distortion function and limit

We start by measuring the difference between *standard* and *early* distortion cost.[7] Figure 1 shows the results in terms of BLEU and KRS, plotted against the distortion limit (DL).

Indeed, early distortion cost (Moore and Quirk, 2007) outperforms the standard one in all the tested configurations and according to both metrics. We can see that the quality of both systems deteriorates as the distortion limit increases, however the system with early distortion cost is more robust to this effect. In particular, when passing from DL=12 to DL=18, the baseline system loses 1.2 BLEU and no less than 6.8 KRS, whereas the system with early distortion cost loses 0.8 BLEU and 4.9 KRS. Given these results, we decide to use early distortion cost in all the remaining experiments.

6.3 WaW reordering pruning

We have seen that early distortion cost can effectively reduce the loss of translation quality, but cannot totally prevent it. Moreover, increasing the distortion limit means exploring many more

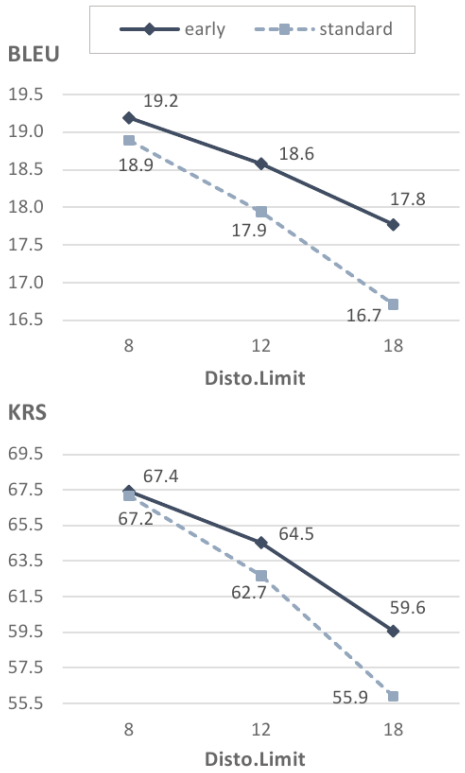

Figure 1: Standard *vs* early distortion cost performance measured in terms of BLEU and KRS on tests(09-11) under different distortion limits.

hypotheses and, consequently, slowing down the decoding process. With our WaW model-based reordering pruning technique, we aim at solving both issues.

We generate the WaW training data from the first 30K sentences of the News-commentary-2010 parallel corpus, using a sampling window of width δ=10. This results in 8 million training samples, which are fed to the binary classifier implementation of the MegaM Toolkit[8]. Features with less than 20 occurrences are ignored and the maximum number of training iterations is set to 100.

Evaluated intrinsically on test08, the model achieves the following classification accuracy: 67.0% precision, 50.0% recall, 57.2% F-score. While these figures are rather low, we recall that the WaW model is not meant to be used as a standalone classifier, but rather as one of several SMT feature functions and as a way to detect very unlikely reordering steps. Hence, we also evaluate its ability to rank a typical set of reordering options during decoding: that is, we traverse the source words in target order and, for each of them, we ex-

[7] For this first series of experiments, feature weights are tuned in the DL=8 setting and the two resulting weight vectors (one for standard, one for early distortion) are re-used in the higher-DL experiments.

[8] http://www.cs.utah.edu/~hal/megam/ (Daumé III, 2004).

System	DL	tests(09-11) bleu	met	krs	krs-V	test12 bleu	met	krs	krs-V	ms/ word
Allowing only short to medium-range reordering:										
PSMT, early disto	8	19.2	28.1	67.4	65.4	19.0	28.1	67.8	66.1$^{\blacktriangle}$	202
+WaW (feature only)		19.4$^{\blacktriangle}$	28.2$^{\blacktriangle}$	67.6$^{\blacktriangle}$	65.5$^{\blacktriangle}$	19.5$^{\blacktriangle}$	28.3$^{\blacktriangle}$	67.8	66.2	212
HSMT, max.span=10		20.1$^{\blacktriangle}$	28.5$^{\blacktriangle}$	68.4$^{\blacktriangle}$	66.7$^{\blacktriangle}$	19.7$^{\triangle}$	28.4$^{\triangle}$	68.6$^{\blacktriangle}$	67.3$^{\blacktriangle}$	406
Allowing also long-range reordering:										
PSMT, early disto	18	18.2	28.0	62.9	62.0	18.2	28.1	63.4	62.5	408
+WaW (feature only)		18.4$^{\blacktriangle}$	28.0	61.8$^{\blacktriangledown}$	61.3$^{\blacktriangledown}$	18.1	28.1	62.2$^{\blacktriangledown}$	61.7$^{\blacktriangledown}$	428
+WaW reo.pruning (ϑ=5)		19.5$^{\blacktriangle}$	28.3$^{\blacktriangle}$	67.9$^{\blacktriangle}$	66.3$^{\blacktriangle}$	19.3$^{\blacktriangle}$	28.4$^{\blacktriangle}$	67.8$^{\blacktriangle}$	66.3$^{\blacktriangle}$	**142**
HSMT, max.span=20		20.0$^{\blacktriangle}$	28.5$^{\blacktriangle}$	68.1$^{\blacktriangle}$	66.7$^{\blacktriangle}$	19.7$^{\blacktriangle}$	28.4	68.2$^{\blacktriangle}$	67.1$^{\blacktriangle}$	706

Table 1: Effects of WaW reordering model and early reordering pruning on PSMT translation quality and efficiency, compared against a hierarchical SMT baseline. Translation quality is measured with % BLEU, METEOR, and Kendall Reordering Score: regular (KRS) and verb-specific (KRS-V). Statistically significant differences with respect to the previous row are marked with $^{\blacktriangle\blacktriangledown}$ at the $p \leq .05$ level and $^{\triangle\triangledown}$ at the $p \leq .10$ level. Decoding time is measured in milliseconds per input word.

amine the ranking of all words that may be translated next (i. e. the uncovered positions within a given DL). We find that, even when the DL is very high (18), the correct jump is ranked among the top 3 reachable jumps in the large majority of cases (81.4%). If we only consider long jumps – i. e. spanning more than 6 words – the Top-3 accuracy is 56.4% while that of a baseline that simply favors shorter jumps (as the distortion cost does) is only 26.5%.

For the early reordering pruning experiment, we set the pruning parameters to 2 for histogram and 0.25 for relative threshold.[9] A non-prunable-zone of width ϑ=5 is set around the last covered position. The resulting configuration is re-optimized by MERT on test08 for the final experiment.

Table 1 shows the effects of integrating the WaW reordering model into a PSMT decoder that already includes a state-of-the-art hierarchical phrase orientation model. The same table also presents the results of the HSMT constrastive experiments. Two scenarios are considered: in the first block, the PSMT distortion limit is set to a medium value (8) and the HSMT maximum span constraint is set to 10. Although not directly comparable, these settings have the same effect of disallowing long-range reorderings. In the second block, long-range reorderings are instead allowed with a DL of 18 and a HSMT span constraint of 20.

Feature weights are optimized for each experiment using the procedure described above (four averaged MERT runs). Statistical significance is computed for each experiment against the previous one (i. e. previous row), using approximate randomization as in (Riezler and Maxwell, 2005). Run times are obtained by an Intel Xeon X5650 processor on the first 500 sentences of tests(09-11), excluding loading time of all models.

Medium reordering space. Integrating the WaW model as an additional feature function yields small but consistent improvements (second row of Table 1). Concerning the run time, we notice just a small overload of about 5%: that is, from 202 to 212 ms/word.

In comparison, the tree-based system (third row) has almost double decoding time but achieves statistically significant higher translation quality, especially at the level of reordering.

Large reordering space. As expected, raising the DL to 18 with no special pruning (fourth row) results in much slower decoding (from 202 to 408 ms/word) but also in very poor translation quality. This loss is especially visible on the reordering scores: e. g. from 67.4 to 62.9 KRS on tests(09-11). Unfortunately, adding the WaW model as a feature function (fifth row) does not appear to be helpful under the high DL condition.

On the other hand, when using the WaW model

[9] Pruning parameters were optimized for BLEU with a grid search over the values (1, 2, 3, 4, 5) for histogram and (0.5, 0.25, 0.1) for threshold.

	adv.	verb_{mod}	subj.	obj.	compl.
SRC	Jedoch	**konnten**	sie	Kinder in Teilen von Helmand und Kandahar im Süden	aus Sicherheit∼ grund
	however	*could*	*they*	*children in parts of Helmand and Kandahar in South*	*for security reasons*
(de)	neg	verb_{inf}			
	nicht	**erreichen** .			
	not	*reach*			
REF	But they **could not reach** children in parts of Helm. and Kand. in the south for security reasons.				
BASE-8	However, they **were** children in parts of Helm. and Kand. in the south, for security reasons.				
HIER-10	However, they **were** children in parts of Helm. and Kand. in the south **not reach** for security reasons.				
BASE-18	However, they **were** children in parts of Helm. and Kand. in the south **do not reach** for security reasons.				
WAWP-18	However, they **could not reach** children in parts of Helm. and Kand. in the south for security reasons.				
HIER-20	However, they **were** children in parts of Helm. and Kand. in the south **not reach** for security reasons.				

Table 2: Long-range reordering example showing the behavior of different systems: [BASE-*] are phrase-based systems with a DL of 8 and 18 respectively; [WAWP-18] refers to the WaW-pruning PSMT system; [HIER-*] are hierarchical SMT systems with a span constraint of 10 and 20 words respectively.

also for reordering pruning (sixth row) we are able to recover the performance of the medium-DL baseline performance and even to slightly improve it. It is interesting to note that the largest improvement concerns the accuracy of verb reordering on tests(09-11): from 65.4 to 66.3 KRS-V. Although the other gains are rather small, we emphasize the fact that our solutions mostly affect rare and isolated events, which have a limited impact on the general purpose evaluation metrics but are are essential to produce readable translations. WaW reordering pruning has also a remarkable effect on efficiency, making decoding time decrease from 428 ms/word to 142 ms/word, that is even faster than a baseline that does not explore any long-range reordering at all (202 ms/word).

Finally, we can see from the last row of Table 1 that the gap between PSMT and HSMT has been narrowed significantly. While more work is needed to reach and outperform the quality of the HSMT system, we were able to closely approach it with five times lower decoding time (142 versus 706 ms/word) and about ten times smaller models (cf. Section 5). Comparing our best system with the best HSMT system (i. e. span constraint 10), we see that the gap in translation accuracy is slightly larger and that the decoding speed-up is smaller (142 versus 406 ms/word). However, the better performance and efficiency of HSMT-10 comes at the expense of all long-range reorderings.

Thus, our enhanced PSMT appears as an optimal choice in terms of trade-off between translation quality and efficiency.

Table 3 reports two kinds of decoding statistics that allow us to explain the very different decoding times observed, and to verify that the WaW-pruning system actually performs long-range reorderings: **#hyp/sent** is the average number of partial translation hypotheses created[10] per test sentence; **(#jumps/sent)×100** is the average number of phrase-to-phrase jumps included in the 1-best translation of every 100 test sentences. Only medium and long jumps are shown (distortion D≥6), divided into three distortion buckets.

System	DL	#hyp/sent	(#jumps/sent)×100 D: [6..8]	[9..12]	[13..18]
baseline	8	600K	90	–	–
baseline	18	1278K	88	61	48
+WaW r.prun.	18	364K	52	29	17

Table 3: Decoding statistics of three PSMT systems exploring different reordering search spaces for the translation of test12.

We can see that the early-pruning system indeed performed several long jumps but it explored a much smaller search space compared to the high-distortion baseline (364K versus 1278K partial hypotheses). As for the lower number of long jumps (e. g. 29 versus 61 with D in [9..12] and 17 versus 48 in [13..18]) it suggests that the early-pruning system is more precise, while the high-distortion baseline is over-reordering.

The output of different systems for our example sentence is shown in Table 2. In this sentence, a jump forward with D=12 and a jump backward with D=14 were necessary to achieve the correct reordering of the verb and its negation. Although

[10] That is, the hypotheses that were scored by all the PSMT model components and added to a hypothesis stack.

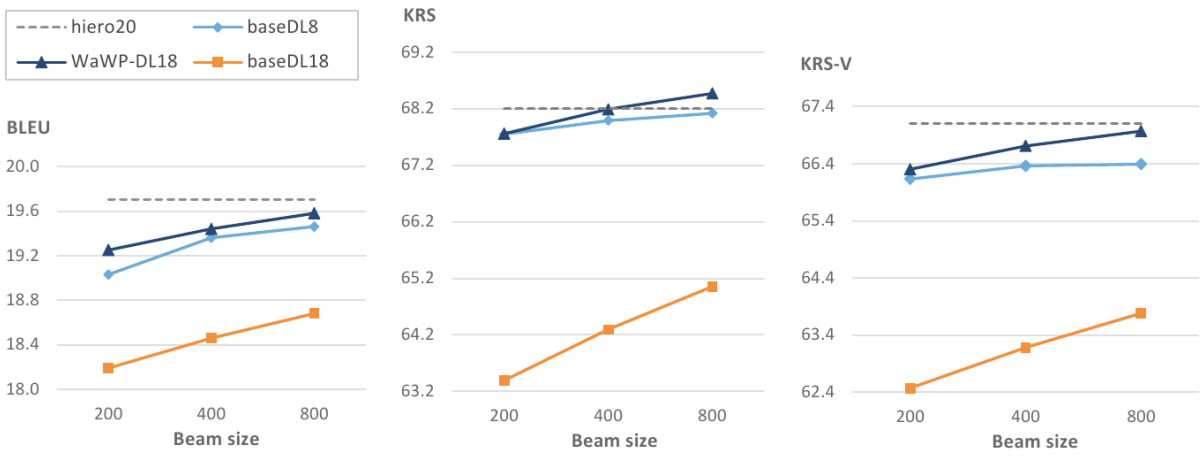

Figure 2: Effects of beam size on translation quality measured by BLEU, KRS and KRS-V, in two baseline PSMT systems (DL=8 and DL=18) and in the WaW early-pruning system (test12). For comparison, the hierarchical system performance (span constraint 20) is provided as a dotted line.

these jumps were reachable for both the [PSMT-18] and the [HSMT-20] systems, only the WaW-pruning PSMT system actually performed them.

6.4 Interaction with beam-search pruning

During the beam-search decoding process, early reordering pruning interacts with regular hypothesis pruning based on the weighted sum of all model scores. In particular, all the PSMT systems presented so far apply a default histogram threshold of 200 to each hypothesis stack. To examine this interaction, we increase the histogram threshold (beam size) from the default value of 200 up to 800, while keeping all other parameters and feature weights fixed. The results on test12 are plotted against the beam size and reported in Figure 2. The dotted line in each plot represents the performance of the hierarchical system presented in the last row of Table 1 (span constraint 20).

We can see that increasing the beam size is more beneficial for the high-DL baseline (baseDL18) than for the medium one (baseDL8). This is not surprising as the risk of search error is higher when a larger search space is explored with equal models and pruning parameters. Nevertheless, baseDL18 remains by far the worst performing system, even in our largest beam setting (800) corresponding to four times longer decoding time (1582 ms/word). What is remarkable, instead, is that the larger beam size also results in better performances by the WaW-pruning system, which is the PSMT system that explores by far the smallest search space (cf. Table 3). The superiority of the WaW-pruning system over the PSMT baselines is maintained in all tested settings and according to all metrics, which confirms the usefulness of our methods not only as optimization techniques, but also for reducing model errors of a baseline that already includes strong reordering models.

With a very large beam size (800) our enhanced PSMT system can closely approach the performance of HSMT-20 in terms of BLEU and KRS-V, and even surpass it in terms of KRS (statistically significant) while still remaining faster: that is, 554 versus 706 ms/word.

Overall HSMT-10 remains the best system, with slightly higher KRS and KRS-V and lower decoding time than our best enhanced PSMT system (406 versus 554 ms/word). However, we note once more that this performance comes at the expense of all long-range reorderings. For a completely fair comparison, the HSMT system should also be enhanced with similar reordering-pruning techniques – a research path that we plan to explore in the future, possibly inspiring from the approach of Braune et al. (2012).

7 Conclusions

We have presented a few techniques that can improve the accuracy of the word reordering performed by a German-English phrase-based SMT system. In particular, we have shown how long-range reorderings can be captured without worsening the general quality of translation and without renouncing to efficiency. Our best PSMT system is actually faster than a system that does not even attempt to perform long-range reordering, and it

obtains significantly higher evaluation scores.

In comparison to a more computationally costly tree-based approach (hierarchical SMT), our enhanced PSMT system produces slightly lower translation quality but in five times lower decoding time when long-range reordering is allowed. Moreover, when a larger beam size is explored, the performance of our system can equal that of the long-reordering hierarchical system, but still with faster decoding.

In summary, we have shown that an appropriate modeling of the word reordering problem can lead to narrow or even fill the gap between phrase-based and hierarchical SMT in this difficult language pair. We have also disproved the common belief that sacrificing long-range reorderings by setting a low distortion limit is the only way to obtain well-performing PSMT systems.

Acknowledgments

This work was partially funded by the European Union under FP7 grant agreement EU-BRIDGE, Project Number 287658.

References

Yaser Al-Onaizan and Kishore Papineni. 2006. Distortion models for statistical machine translation. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics*, pages 529–536, Sydney, Australia, July.

Satanjeev Banerjee and Alon Lavie. 2005. METEOR: An automatic metric for MT evaluation with improved correlation with human judgments. In *Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*, pages 65–72, Ann Arbor, Michigan, June.

Alexandra Birch, Miles Osborne, and Philipp Koehn. 2008. Predicting success in machine translation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 745–754, Stroudsburg, PA, USA.

Alexandra Birch, Miles Osborne, and Phil Blunsom. 2010. Metrics for MT evaluation: evaluating reordering. *Machine Translation*, 24(1):15–26.

Arianna Bisazza and Marcello Federico. 2013. Dynamically shaping the reordering search space of phrase-based statistical machine translation. To appear in *Transactions of the ACL*.

Arianna Bisazza. 2013. *Linguistically Motivated Reordering Modeling for Phrase-Based Statistical Machine Translation*. Ph.D. thesis, University of Trento. http://eprints-phd.biblio.unitn.it/1019/.

Fabienne Braune, Anita Gojun, and Alexander Fraser. 2012. Long-distance reordering during search for hierarchical phrase-based SMT. In *Proceedings of the Annual Conference of the European Association for Machine Translation (EAMT)*, pages 28–30, Trento, Italy.

Chris Callison-Burch, Philipp Koehn, Christof Monz, Matt Post, Radu Soricut, and Lucia Specia. 2012. Findings of the 2012 workshop on statistical machine translation. In *Proceedings of the Seventh Workshop on Statistical Machine Translation*, pages 10–51, Montréal, Canada, June.

Mauro Cettolo, Nicola Bertoldi, and Marcello Federico. 2011. Methods for smoothing the optimizer instability in SMT. In *MT Summit XIII: the Thirteenth Machine Translation Summit*, pages 32–39, Xiamen, China.

Stanley F. Chen and Joshua Goodman. 1999. An empirical study of smoothing techniques for language modeling. *Computer Speech and Language*, 4(13):359–393.

David Chiang. 2005. A hierarchical phrase-based model for statistical machine translation. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL'05)*, pages 263–270, Ann Arbor, Michigan, June.

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

Michael Collins, Philipp Koehn, and Ivona Kucerova. 2005. Clause restructuring for statistical machine translation. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL'05)*, pages 531–540, Ann Arbor, Michigan, June.

Hal Daumé III. 2004. Notes on CG and LM-BFGS optimization of logistic regression. Paper available at `http://pub.hal3.name`, implementation available at `http://hal3.name/megam`.

Marcello Federico, Nicola Bertoldi, and Mauro Cettolo. 2008. IRSTLM: an Open Source Toolkit for Handling Large Scale Language Models. In *Proceedings of Interspeech*, pages 1618–1621, Brisbane, Australia.

Minwei Feng, Arne Mauser, and Hermann Ney. 2010. A source-side decoding sequence model for statistical machine translation. In *Conference of the Association for Machine Translation in the Americas (AMTA)*, Denver, Colorado, USA.

Michel Galley and Christopher D. Manning. 2008. A simple and effective hierarchical phrase reordering model. In *EMNLP '08: Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 848–856, Morristown, NJ, USA.

Dmitriy Genzel. 2010. Automatically learning source-side reordering rules for large scale machine translation. In *Proceedings of the 23rd International Conference on Computational Linguistics*, COLING '10, pages 376–384, Stroudsburg, PA, USA.

Anita Gojun and Alexander Fraser. 2012. Determining the placement of German verbs in English-to-German SMT. In *Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics*, pages 726–735, Avignon, France, April.

Spence Green, Michel Galley, and Christopher D. Manning. 2010. Improved models of distortion cost for statistical machine translation. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics (NAACL)*, pages 867–875, Los Angeles, California.

Hieu Hoang and Philipp Koehn. 2010. Improved translation with source syntax labels. In *Proceedings of the Joint Fifth Workshop on Statistical Machine Translation and MetricsMATR*, pages 409–417, Uppsala, Sweden, July.

Hieu Hoang, Philipp Koehn, and Adam Lopez. 2009. A unified framework for phrase-based, hierarchical, and syntax-based statistical machine translation. In *International Workshop on Spoken Language Translation (IWSLT)*, pages 152–159, Tokyo, Japan.

H. Johnson, J. Martin, G. Foster, and R. Kuhn. 2007. Improving translation quality by discarding most of the phrasetable. In *In Proceedings of EMNLP-CoNLL 07*, pages 967–975.

Philipp Koehn, Franz Josef Och, and Daniel Marcu. 2003. Statistical phrase-based translation. In *Proceedings of HLT-NAACL 2003*, pages 127–133, Edmonton, Canada.

Philipp Koehn, Amittai Axelrod, Alexandra Birch Mayne, Chris Callison-Burch, Miles Osborne, and David Talbot. 2005. Edinburgh system description for the 2005 IWSLT speech translation evaluation. In *Proc. of the International Workshop on Spoken Language Translation*, October.

P. Koehn, 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 *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics Companion Volume Proceedings of the Demo and Poster Sessions*, pages 177–180, Prague, Czech Republic.

Kimmo Koskenniemi and Mariikka Haapalainen, 1994. *GERTWOL – Lingsoft Oy*, chapter 11, pages 121–140. Roland Hausser, Niemeyer, Tübingen.

Percy Liang, Ben Taskar, and Dan Klein. 2006. Alignment by agreement. In *Proceedings of the Human Language Technology Conference of the NAACL, Main Conference*, pages 104–111, New York City, USA, June.

Robert C. Moore and Chris Quirk. 2007. Faster beam-search decoding for phrasal statistical machine translation. In *In Proceedings of MT Summit XI*, pages 321–327, Copenhagen, Denmark.

Jan Niehues and Muntsin Kolss. 2009. A POS-based model for long-range reorderings in SMT. In *Proceedings of the Fourth Workshop on Statistical Machine Translation*, pages 206–214, Athens, Greece.

F. Och and H. Ney. 2002. Discriminative training and maximum entropy models for statistical machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 295–302, Philadelhpia, PA.

Franz Josef Och. 2003. Minimum Error Rate Training in Statistical Machine Translation. In Erhard Hinrichs and Dan Roth, editors, *Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics*, pages 160–167.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2001. Bleu: a method for automatic evaluation of machine translation. Research Report RC22176, IBM Research Division, Thomas J. Watson Research Center.

Stefan Riezler and John T. Maxwell. 2005. On some pitfalls in automatic evaluation and significance testing for MT. In *Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*, pages 57–64, Ann Arbor, Michigan, June.

Helmut Schmid. 1994. Probabilistic part-of-speech tagging using decision trees. In *Proceedings of International Conference on New Methods in Language Processing*.

Christoph Tillmann. 2004. A Unigram Orientation Model for Statistical Machine Translation. In *Proceedings of the Joint Conference on Human Language Technologies and the Annual Meeting of the North American Chapter of the Association of Computational Linguistics (HLT-NAACL)*.

Roy Tromble and Jason Eisner. 2009. Learning linear ordering problems for better translation. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing*, pages 1007–1016, Singapore, August.

Karthik Visweswariah, Rajakrishnan Rajkumar, Ankur Gandhe, Ananthakrishnan Ramanathan, and Jiri Navratil. 2011. A word reordering model for improved machine translation. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*, pages 486–496, Edinburgh, Scotland, UK., July.

Fei Xia and Michael McCord. 2004. Improving a statistical MT system with automatically learned rewrite patterns. In *Proceedings of Coling 2004*, pages 508–514, Geneva, Switzerland, Aug 23–Aug 27. COLING.

R. Zens, F. J. Och, and H. Ney. 2002. Phrase-based statistical machine translation. In *25th German Conference on Artificial Intelligence (KI2002)*, pages 18–32, Aachen, Germany. Springer Verlag.