Mach Translat (2006) 20:199–215
DOI 10.1007/s10590-007-9024-z

Improving statistical MT by coupling reordering and decoding

Josep Maria Crego · José B. Mariño

Received: 11 October 2006 / Accepted: 24 May 2007 / Published online: 12 July 2007

Abstract In this paper we describe an elegant and efficient approach to coupling reordering and decoding in statistical machine translation, where the n-gram translation model is also employed as distortion model. The reordering search problem is tackled through a set of linguistically motivated rewrite rules, which are used to extend a monotonic search graph with reordering hypotheses. The extended graph is traversed in the global search when a fully informed decision can be taken. Further experiments show that the n-gram translation model can be successfully used as reordering model when estimated with reordered source words. Experiments are reported on the Europarl task (Spanish–English and English–Spanish). Results are presented regarding translation accuracy and computational efficiency, showing significant improvements in translation quality with respect to monotonic search for both translation directions at a very low computational cost.

Keywords Statistical MT · Reordering · Decoding · n-gram language models

1 Introduction

In classical statistical machine translation (SMT), each source sentence s_1^J is transformed into (or generates) a target sentence t_1^I, by means of a stochastic process. Thus, translation of a source sentence s_1^J can be formulated as the search of the target sentence t_1^I that maximizes the conditional probability $p(t_1^I|s_1^J)$, which can be rewritten

J. M. Crego (✉) · J .B. Mariño
Department of Signal Theory and Communications, TALP Research Center, Universitat Politècnica de Catalunya, Campus Norte UPC, Edificio D5, C/Jordi Girona 1-3, 08034 Barcelona, Spain
e-mail: jmcrego@gps.tsc.upc.edu

J .B. Mariño
e-mail: canton@gps.tsc.upc.edu

using the Bayes rule as (1).

$$\hat{t}_1^I = \arg\max_{t_1^I} \left\{ p(s_1^J | t_1^I) \cdot p(t_1^I) \right\} \tag{1}$$

The first SMT systems worked at the word level (Brown et al. 1990). In this first system, differences in word order between source and target languages made reordering a very hard problem in terms of both modeling and decoding. In Knight (1999), the search problem is classified NP-complete when arbitrary word reorderings are permitted, while polynomial time search algorithms can be obtained under monotonic conditions.

The source channel approach in (1) is nowadays replaced by a maximum entropy framework (Berger et al. 1996) that makes it easier to introduce additional models (Och and Ney 2002) (2).

$$\hat{t}_1^I = \arg\max_{t_1^I} \left\{ \sum_{m=1}^{M} \lambda_m h_m(s_1^J, t_1^I) \right\} \tag{2}$$

In (2), λ_m corresponds to the weighting coefficients of the log-linear combination, and the feature functions $h_m(s, t)$ correspond to a logarithmic scaling of the probabilities of each model. Coefficients are typically optimized to maximize a scoring function (Och 2003).

The appearance of phrase-based (in contrast to word-based) translation models brought a clear improvement in the state of the art of SMT (Zens et al. 2002). The phrase-based approach introduced bilingual phrases (contiguous sequence of words in both languages) as translation units which naturally capture local reorderings, thus alleviating the reordering problem.

However, the phrase-based approach did not entirely solve the reordering problem, showing a main weakness on longest reorderings which are only tackled by using long phrases, not always present in the training corpus because of the obvious data sparseness problem.

In recent years huge research efforts have been conducted aiming at developing improved reordering approaches. In the next section several of the proposed alternatives are discussed.

1.1 Antecedents

The first SMT systems introducing reordering capabilities were founded on the brute force of computers, aiming at finding the best hypothesis through traversing a fully reordered search graph (all permutations of source-side words are allowed in the search). This approach is computationally very expensive, even for very short input sentences. Hence, in order to make the search feasible, several reordering constraints have been developed:

- "IBM". Each new target word must be aligned to one of the first k uncovered source words (Brown et al. 1993).

- "Local". A given source word is allowed to be reordered only k positions distant from its original position (Kanthak et al. 2005).
- "MaxJumps". The number of reorderings for a search path (whole translation) is limited to a given number (Crego et al. 2005a).
- "ITG" (Inversion Transduction Grammars) (Wu 1996). The input sentence is seen as a sequence of blocks, and a pair of blocks are merged by either keeping the monotonic order (original) or inverting their order. This constraint is founded on the parse trees of the simple grammar in Wu (1997).

The use of these constraints implies a necessary balance between translation accuracy and efficiency.

Typically, a distance-based reordering model is used during the search to penalize longest reorderings, only allowed when well supported by the rest of the model. Additionally, lexicalized reordering models have been introduced which score reorderings in search using distance between words seen in training (Tillmann 2004; Kumar and Byrne 2005), distance between phrase pairs (Tillmann and Zhang 2005; Nagata et al. 2006), based on adjacency/swap of phrases (Koehn et al. 2005), and using POS tags, lemmas and word classes to gain generalization power (Zens and Ney 2006).

A main criticism to this brute-force approach is the lack of linguistic information used to limit the search graph, while in linguistic theory, reorderings between linguistic phrases are well described.

Current SMT systems tend to introduce linguistic information into new reordering strategies to overcome the efficiency problem. Several alternatives have been proposed:

- Adding a word-order monotonization task before the global search, consisting of learning reorderings into the source side to achieve a similar word order to that of the target side (Xia and McCord 2004; Collins et al. 2005; Costa-jussà and Fonollosa 2006; Popovic and Ney 2006).
- Extending standard phrases to account for "holes" in either the target or the source side (Simard et al. 2005).
- Using syntax (structure) information that is incorporated into the SMT system in different ways:
  - Using standard phrases extended with syntax information of the source side, though using dependency trees (Langlais and Gotti 2005; Quirk et al. 2005).
  - Building translations as derivations (syntax-directed translations), exploiting the power of synchronous rewriting systems. These systems use source- and/or target-constituent trees (instead of dependency trees), which can be formally syntax-based (Chiang 2005; Watanabe et al. 2006) or linguistically syntax-based (Wu 1997; Yamada and Knight 2002).

We propose a reordering framework where differences in word order between language pairs are harmonized in training using word-to-word alignments and learned in the form of reordering patterns (Crego and Mariño 2006a). Patterns are built using POS tags in order to acquire generalization power. In decoding, reordering hypotheses are proposed following the previous rules. Therefore, the monotonic search is slightly extended with linguistically motivated reorderings. Furthermore, the n-gram translation model is successfully used to account for the source word order pursued in decoding, as it has been learned with reordered source words. Finally, a fully informed reordering decision is taken in consensus by the whole SMT model (Crego and Mariño 2006b).

The paper is organized as follows. In Sect. 2 we briefly review the particularities of the translation system used in this work. Section 3 details the reordering framework proposed. Section 4 reports the experiments conducted to assess the accuracy/efficiency of the framework, and finally, Sect. 5 concludes and outlines further work.

2 SMT System

Our SMT system follows the maximum entropy framework (Berger et al. 1996) presented in (2). Following this approach, the *baseline* translation system implements a log-linear combination of one translation model and five additional feature functions (models):

- The *translation model* is expressed in tuples as translation (or bilingual) units (Crego et al. 2004).
  Given a word-to-word alignment, tuples define a unique and monotonic segmentation of each bilingual sentence, building up a much smaller set of units than with standard phrases and allowing n-gram estimation to account for the history of the translation process (Mariño et al. 2006). Figure 1 shows an example of a sentence pair segmented into four units (tuples). Equation (3) describes the particular n-gram language model,

$$p_{\mathrm{TM}}(s_1^J, t_1^I) = \left\{ \prod_{i=1}^{K} p((s,t)_i | (s,t)_{i-N+1}, \ldots, (s,t)_{i-1}) \right\} \tag{3}$$

  where $(s,t)_i$ refers to the ith tuple of a given bilingual sentence pair which is segmented into K units.
- A *target-language model*, estimated as an n-gram language model over the target words (4),

$$p_{\mathrm{LM}}(s_1^J, t_1^I) \approx \prod_{i=1}^{I} p(t_i | t_{i-N+1}, \ldots, t_{i-1}) \tag{4}$$

  where t_i refers to the ith target word.
- A *word-bonus model* used in order to compensate the system preference for short target sentences caused by the presence of the previous target language model (5).

$$p_{\mathrm{WB}}(s_1^J, t_1^I) = \exp(I) \tag{5}$$

- A *source-to-target* and a *target-to-source* lexicon model, using IBM model-1 translation probabilities to compute a lexical weight for each tuple, which accounts for

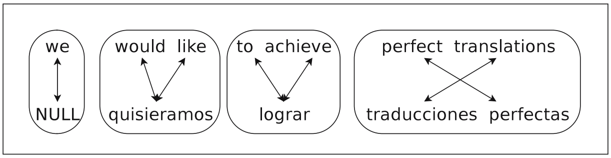

Fig. 1 Segmentation into tuples of a word-to-word aligned sentence pair

the statistical consistency of the pair of words inside the tuple (6–7),

$$p_{\text{IBM1}}((s,t)_n) = \frac{1}{(I+1)^J} \prod_{j=1}^{J} \sum_{i=0}^{I} p(t_n^i | s_n^j) \tag{6}$$

$$p_{\text{IBM1}'}((s,t)_n) = \frac{1}{(J+1)^I} \prod_{i=1}^{I} \sum_{j=0}^{J} p(s_n^j | t_n^i) \tag{7}$$

where $(s,t)_n$ refers to the nth unit of a given bilingual sentence pair which is segmented into tuples.

- A *tagged target-language model*, estimated as an n-gram language model over the same target side of the training corpus but using POS tags instead of raw words (8).

$$p_{\text{posLM}}(s_1^J, t_1^I) \approx \prod_{i=1}^{I} p(\text{POS}_i | \text{POS}_{i-N+1}, \ldots, \text{POS}_{i-1}) \tag{8}$$

Given the combination of models presented above, we have used MARIE, the freely available decoder implementing a beam-search strategy with distortion (or reordering) capabilities (Crego 2005; Crego et al. 2005a).

Further details of the SMT system are given in Sect. 4.

3 Reordering

In this section we describe the reordering framework presented in this work.

3.1 Reordering patterns using POS tags

A reordering pattern consists of the rewrite rule $t_1, \ldots, t_n \mapsto i_1, \ldots, i_n$, where $t_1, \ldots, t_n$ is a sequence of POS tags (related to a sequence of source words), and indices $i_1, \ldots, i_n$ represent a sequence of positions into which the source words are to be reordered.

To extract patterns from the training corpus we use the crossed links found in translation tuples. Patterns can be seen as the reordering rules that applied over the source words of a tuple to generate the word order of the tuple target words.

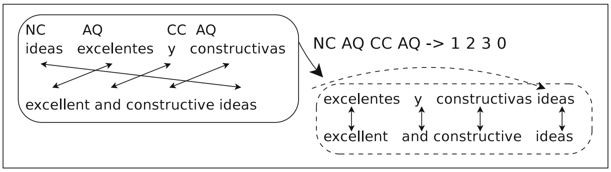
NC AQ CC AQ
ideas excelentes y constructivas
excellent and constructive ideas
NC AQ CC AQ -> 1 2 3 0
excelentes y constructivas ideas
excellent and constructive ideas

Fig. 2 Pattern extraction

Figure 2 shows an example of pattern extraction NC AQ CC AQ $\mapsto$ 1 2 3 0, where the first source word 0, which has been POS tagged NC, is mapped into the last position: 1 2 3 0. The pattern is obtained using word-to-word alignments and the source-side POS tags of a given tuple. As can be seen, the word alignment is monotonized (dashed box) when the pattern is applied over the tuple source words.

Each pattern is scored with a probability computed on the basis of relative frequency (9).

$$p(t_1, \ldots, t_n \mapsto i_1, \ldots, i_n) = \frac{N(t_1, \ldots, t_n \mapsto i_1, \ldots, i_n)}{N(t_1, \ldots, t_n)} \tag{9}$$

This score is used in this work only to filter the set of patterns to be used in decoding.

3.2 Input search graph extension

In decoding, the input sentence is handled as a word graph where a given hypothesis is extended by means of covering (translating) some uncovered source word. However, a monotonic search graph contains a single path, composed of arcs covering the input words in the original word order. To allow for reordering, the graph is extended with new arcs, covering the source words in the desired word order.

The motivation for extending the input graph is double: first, the aim to improve the translation quality is met by the ability of reordering following the patterns as explained previously. Second, the reordering decision is more informed since it is taken during decoding using all the SMT models.

The extension procedure is outlined as follows: starting from the monotonic graph, any sequence of the input POS tags fulfilling a source-side rewrite rule implies the addition of a reordering path (composed of one or more arcs). The reordering path encodes the reordering detailed in the target side of the rule, and is composed of as many arcs as there are words present in the pattern.

Figure 3 shows an example of input search graph extension. Two patterns are found in the example, used to extend the input graph through reordered hypotheses. The first row shows the input sentence (left) and the monotonic search graph (right). In the second row, the search graph is extended with a reordered hypothesis (dotted arcs) following the reordering pattern NC AQ $\mapsto$ 1 0 (where the first two words are swapped). Finally, the third row shows the extension of the search graph following the reordering pattern NC AQ CC AQ $\mapsto$ 1 2 3 0.

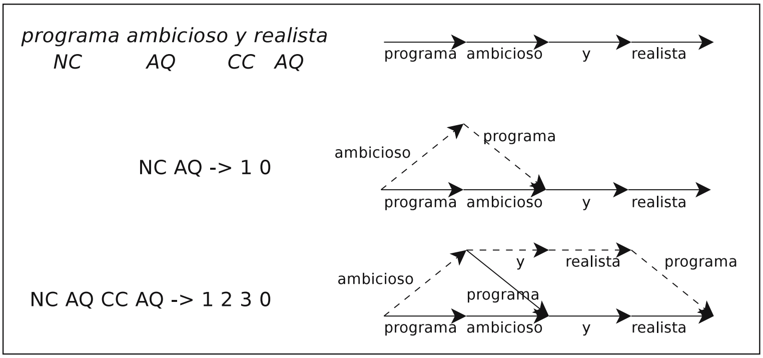

Fig. 3 Input search graph extension

Once the input search graph is built, it is traversed by the decoder aiming at finding the best translation. Hence, the winner hypothesis is computed using the whole set of system models (fully informed decision). The input sentence of the example in Fig. 3 is traversed in decoding, following three different word orders (Ex. 1).

(1) a. *programa ambicioso y realista*
b. *ambicioso programa y realista*
c. *ambicioso y realista programa*

3.3 Reordered *n*-gram translation model

The use of long tuples impoverishes the probability estimates of the translation model, as longer tuples appear less often in training than the smaller ones (data sparseness problem). Therefore, language pairs with significant differences in word order may suffer from poor probability estimates.

Given our special translation model, the problem is specially relevant as translation units (tuples) are learned from a unique segmentation of each training sentence pair, obtaining a smaller number of tuples than phrases are obtained under the phrase-based approach (Crego et al. 2005c).

In Kanthak et al. (2005) and Collins et al. (2005) a procedure prior to the phrase extraction is suggested, aiming at monotonizing the source and target word order of each sentence pair. Following this idea, we propose to estimate the *n*-gram translation model using the “unfold” technique detailed in Crego et al. (2005b) in contrast to the “regular” method detailed in Crego et al. (2004).

The unfolding technique makes use of the word alignments. It can be decomposed into two main steps:

1. First an iterative procedure, where words in one side are grouped when linked to the same word (or group) in the other side. The procedure loops grouping words in both sides until no new groups are obtained.
2. In the second step the resulting groups (unfolded tuples) are output following the word order of target-sentence words. Hence, the tuple sequence modifies the word order of the source sentence.

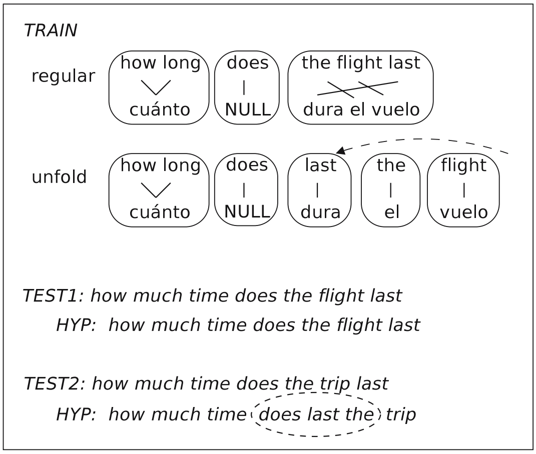
TRAIN
regular
how long
cuánto
does
NULL
the flight last
dura el vuelo
unfold
how long
cuánto
does
NULL
last
dura
the
el
flight
vuelo
TEST1: how much time does the flight last
HYP: how much time does the flight last
TEST2: how much time does the trip last
HYP: how much time does last the trip

Fig. 4 Unfold vs. regular tuple extraction

Figure 4 shows an example of tuple extraction following both techniques. The sequential composition of regular tuples produces the original word order of both source and target sides. Regarding the unfolded tuples, only the original word order of the target words is produced.

The n-gram translation model estimated with unfolded units does not penalize the reorderings seen in training with the same lexical units (seen in training) and can reinforce the context of unfolded tuples as being shorter than regular tuples.

This is illustrated by the example of Fig. 4, where in TEST1 the hypothesis is similarly scored by both models (reordering seen in training), while in TEST2 the model with unfolded tuples scores better than the right hypothesis as it contains the sequence *does#NULL last#dura the#el* already seen in training.

It is worth saying that the estimation of the n-gram translation model with either regular or unfolded tuples does not imply differences in the pattern extraction, which is always performed following regular tuples (as outlined in Sect. 3.1).

4 Experiments

In this section we detail the evaluation framework and report on the experiments carried out.

4.1 Corpus

Transcriptions of sessions of the European Parliament in 22 languages are currently available at the Parliament's website, http://www.europarl.europa.eu/. In the case of the results presented here, we have used the version of this corpus data that was made available by RWTH Aachen University through the TC-STAR consortium.[1]

[1] TC-STAR (Technology and Corpora for Speech to Speech Translation) is a European Community project funded by the Sixth Framework Programme.

Table 1 TC-STAR English–Spanish parallel corpus

	Sentences	Tokens	Types	POS types	Ref. No.	
Training set	English	1.28 m	34.9 m	~106 k	44	–
	Spanish	1.28 m	36.6 m	~153 k	328	–
Development set	English	735	18,764	3,193	41	2
	Spanish	430	15,332	3,217	181	2
Test set	English	1,094	26,917	3,958	42	2
	Spanish	840	22,774	4,081	196	2

Table 1 presents some basic statistics of the training, development and test sets, for each language considered, English and Spanish. More specifically, the statistics presented in Table 1 are the total number of sentences, the total number of words (tokens), the vocabulary size or total number of distinct words (types) and distinct POS tags, and the reference number for each data set.

4.2 System details

The training data was preprocessed using standard tools for tokenizing and filtering. Afterwards, word-to-word alignments were performed in both alignment directions using GIZA++ (Och and Ney 2000), and the union set of both alignments was computed.

Then, two tuple sets for each translation direction were extracted from the union set of alignments (following the regular and unfold techniques). The resulting tuple vocabularies were pruned out considering the N best translations for each tuple source side ($N = 30$ for the English–Spanish direction and $N = 20$ for Spanish–English) in terms of occurrences.

The English side of the training corpus was POS tagged using the freely available TnT tagger (Brants 2000) and for the Spanish side we used the freely available FreeLing tool (Carreras et al. 2004).

While English presents a vocabulary of 44 POS tags, Spanish has a vocabulary of 328 tags. This is due to the fact that Spanish tags include more information on morphology (person, tense, gender, number, and so on). In order to reduce this larger set, the first two characters of each Spanish tag only were used. The first two characters of the Spanish tags contain similar information to that of the English tags.

We used the SRI language modeling toolkit (Stolcke 2002) to compute the n-gram language models, using $n = 4$ and $n = 5$ for the translation and target-language models respectively.

Once the models were computed, optimal log-linear coefficients were estimated for each translation direction and system configuration using an in-house implementation of the widely used downhill simplex method (Nelder and Mead 1965).

The decoder was always set to perform histogram pruning, keeping the best $b = 50$ hypotheses (during the optimization work, histogram pruning was set to keep the best $b = 10$ hypotheses).

Table 2 Spanish–English reordering patterns

Reordering pattern	Example
NC RG AQ CC AQ $\mapsto$ 1 2 3 4 0	*Ideas muy sencillas y elementales*
NC AQ CC AQ $\mapsto$ 1 2 3 0	*Programa ambicioso y realista*
NC AQ RG AQ $\mapsto$ 2 3 1 0	*Control fronterizo más estricto*
NC AQ AQ $\mapsto$ 2 1 0	*Decisiones políticas delicadas*
AQ RG $\mapsto$ 1 0	*Suficiente todavía*
NC AQ $\mapsto$ 1 0	*Decisiones políticas*
JJ CC JJ NN $\mapsto$ 3 0 1 2	Political and symbolic issues
RB JJ JJ NN $\mapsto$ 3 2 0 1	Most suitable financial perspective
JJ JJ NN $\mapsto$ 2 1 0	American occupying forces
RB JJ NN $\mapsto$ 2 0 1	Absolutely rigid control
NN PO JJ $\mapsto$ 2 0 1	Barroso's problems
JJ NN $\mapsto$ 1 0	Italian parliamentarians

Reordering arcs computed for the development and test sets were pruned out when the probability of the corresponding pattern was below a given threshold $\tau_1 = 0.1$ as in (9). Additionally, reordering patterns were computed only for source-side sequences not exceeding a threshold limit of $\tau_2 = 5$ words.

Table 2 shows some examples of the Spanish–English reordering patterns.[2] As can be seen, patterns are very general rules and may be wrong for some examples.

The sequence of tags NC AQ, typically reordered following the pattern NC AQ $\mapsto$ 1 0 may be reordered following different rules when appearing within a longer structure (as in NC AQ CC AQ $\mapsto$ 1 2 3 0). On the other hand, the example *Barroso's problems* is reordered following the pattern NN PO JJ $\mapsto$ 2 0 1, while the correct Spanish word order is 2 1 0, corresponding to the Spanish translation *problemas de Barroso*. In this case, the reordering rule appears because of bad word alignments in training which prevent the correct pattern from being learned, and reduce the usability of the extracted translation units.

Figure 5 illustrates the problem. The link *('s $\rightsquigarrow$ Barroso)* prevents the correct unfolding (upper side). The problem disappears when only the correct alignments are used (lower side). However, the disadvantages of using wrong patterns are reduced because of the fact that translation units are perfectly coupled with the ordering enclosed in patterns.

The framework proposed in this work does not aim at performing perfect reordering decisions before decoding (hard) but only at reducing the number of reorderings that a fully reordered graph performs. The current list of patterns is useful to test the ability of the whole set of models during the global search to discard the wrong paths.

[2] NC, CC, RQ and AQ are Spanish POS tags equivalent to the English POS tags NN, CC, RB and JJ, respectively noun, conjunction, adverb and adjective.

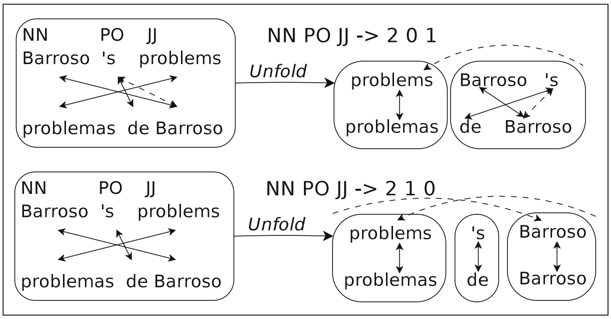

Fig. 5 Wrong pattern extraction because of erroneous word-to-word alignments

Table 3 Translation results over the development and test sets

Direction	Configuration	BLEU (dev)	BLEU	NIST	mWER	PER
Spanish–English	regular + mon	52.61	55.32	10.70	34.28	25.15
	regular + rgraph	53.27	56.14	10.79	33.61	25.23
	unfold + rgraph	53.52	56.11	10.76	33.59	25.31
	unfold + m5j3	51.43	54.47	10.63	34.95	25.56
English–Spanish	regular + mon	47.32	47.75	9.73	41.89	31.84
	regular + rgraph	49.35	49.05	9.92	40.35	31.19
	unfold + rgraph	50.46	50.06	10.00	39.73	30.82
	unfold + m5j3	46.02	47.80	9.81	41.94	31.61

4.3 Results

The algorithms used as evaluation measures were the official TC-STAR evaluation tools distributed by ELDA: BLEU (Papineni et al. 2002), NIST (Doddington 2002), mWER (multi-reference word error rate) and PER (position-independent word error rate).

The first two rows of Table 3 (for both tasks) show translation results under two configurations: "regular + mon" corresponds to a monotonic search using the n-gram translation model built with regular tuples; "regular + rgraph" corresponds to a search allowing for reordering by means of the reordering patterns, and n-gram translation model built with regular tuples.

Confidence intervals for BLEU are ±1.14 for Spanish–English and ±1.44 for English–Spanish.

Results achieved by the "regular + rgraph" configuration are higher than those achieved by the "regular + mon" one. Results are slightly within the confidence interval bounds. However, all measure results are correlated (except for PER, as it does not take into account word reorderings).

The second experiment introduces reordering in the training source words as detailed in Sect. 3.3. The last rows of Table 3 (for both tasks) show translation results under the extended configurations: “unfold + rgraph” corresponds to a search allowing for reordering by means of the reordering patterns, and n-gram translation model built with unfolded tuples; “unfold + m5j3” corresponds to a search allowing for a fully reordered search constrained to a five-word window limit and a maximum of three reorderings per sentence. This configuration introduces a distance-based reordering model in the log-linear combination corresponding to (10),

$$p_{\mathrm{RM}}(t_1^K) = \exp\left(-\sum_{k=1}^{K} d_k\right) \tag{10}$$

where d_k is the distance between the first word of the kth tuple, and the last word +1 of the $(k-1)$th tuple (distances are measured in words referring to the units source side).

Regarding the “unfold + rgraph” configuration, the accuracy level is further improved for the English–Spanish task (achieving clearly statistical significance). The explanation focuses on the n-gram translation model. It is estimated with unfolded tuples implying a smaller set (vocabulary) of translation units of lower size, thereby reducing the sparseness problem and the perplexity of the model. The richer morphology of Spanish makes the sparseness problem more important for the English–Spanish task, which explains why only this task takes advantage of using unfolded units. Again, all measure results are correlated.

Configuration “unfold + m5j3” obtains the worst results. An important bias of the *beam*-based decoders when allowing for reordering is the preference for translating first the easiest parts of the input sentence. Later on in the search, the decoder backtracks to recover older hypotheses which may be pruned out because of the many noisy hypotheses overpopulating the beam.

Table 4 shows a human evaluation of the reordered hypotheses for the English–Spanish test set. Results were computed when setting $\tau_1 = 5$, $\tau_2 = 0.1$ and under the “unfold + rgraph” configuration. This evaluation was performed regarding the paths added to the graph as extended arcs (reorderings), evaluating as erroneous bad ordering decisions. A bad ordering decision is counted when either reordering or keeping the monotonic order is subjectively a wrong decision. By “bad decision” we do not mean a bad translation (as is already noted by the automatic measures) but a bad word order in the target language. For instance, given the input phrase *ambitious and realistic programme* if the decoder decides to use the pattern (JJ CC JJ NN $\mapsto$ 3 0 1 2) showing the translation *programa ambicioso y surrealista*, we count this a success, even if the translation is semantically wrong (the correct word order was achieved).

In Table 4, the column headed “Extended” shows the number of reorderings extending the input graph introduced by the patterns, while “Selected” shows the number of reorderings selected by the decoder and “% error” shows the percentage of subjective errors detected on the sequences of words the decoder was allowed to reorder.

As can be seen, the reordering hypotheses introduced by each pattern have not always been selected in decoding, which results in a limited number of errors. Therefore, we can conclude that the decoder (following the SMT models) has been able to

Table 4 Human evaluation

Pattern size	Extended	Types	Selected	(%)	% error
2	2,617	24	1,088	41.5	4.5
3	1,428	101	266	18.6	6.0
4	1,132	362	143	12.6	6.5
5	892	520	54	6.1	9.0
Total	6,069	1,007	1,551	25.5	6.5

Table 5 Effect of τ_1 and τ_2 on efficiency and translation quality

	Dev		Test	
Threshold	Arcs/sent.	BLEU	Arcs/sent.	BLEU
τ_1 $(\tau_2 = 5)$				
0.01	92.4	49.90	76.5	49.30
0.05	56.5	50.60	47.1	49.91
0.10	47.8	50.46	40.3	50.06
0.20	41.2	50.45	35.0	49.56
0.30	38.3	50.42	32.9	49.79
τ_2 $(\tau_1 = 0.1)$				
5	47.8	50.46	40.3	50.06
4	42.4	50.25	36.6	49.34
3	36.5	49.59	32.8	49.30
$\tau_1 = 1, \tau_2 = 1$	25.5	47.32	24.6	47.75

decide whether a pattern (sometimes a very general rule) was suitable to be used for a given instance or not.

The error rate results of the human evaluation were computed over 800 reordering sequences, 200 for each set of patterns. For all sets, 100 sequences were selected in decoding within the best translation hypothesis.

Table 5 shows the effect on efficiency, in terms of the average number of arcs of the input search graph per sentence, and translation quality in terms of BLEU score produced by the thresholds used in the pattern extraction corresponding to a minimum probability (τ_1) and a maximum size of pattern (τ_2). Results were computed for the English–Spanish task. The last row shows results for a monotonic search.

Regarding threshold results, τ_1=0.1 exhibits a good trade-off between accuracy and efficiency while τ_2 seems to need further experiments with higher values to establish the accuracy limit.

The reason for discarding longer patterns is the sparseness problem appearing in the used rules. The vocabulary of patterns increases exponentially with the size of the rules, while very few examples appear for each. Furthermore, the memory needs of the algorithms are exponentially increased too.

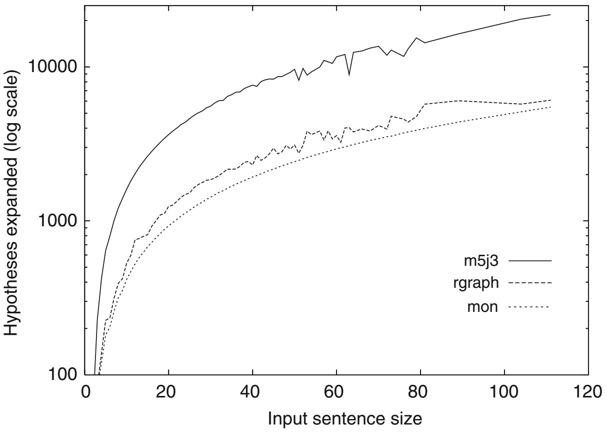

Fig. 6 Global search graph size under different reordering constraints

Longest rules typically respond to reorderings between full (linguistic) phrases, which are not restricted to any size. In order to capture this long-distance reordering new approaches are needed.

Finally, Fig. 6 shows the number of expanded hypotheses (given the input sentence size) under different reordering constraints: "mon" monotonic search condition, "rgraph" allowing for reordering using reordering patterns with unfolded tuples and "m5j3" allowing for full reordering with limiting constraints (maximum number of reorderings per sentence limited to three and maximum reordering distance of five words). Results were computed for the English–Spanish test set.

The search extended with reordering patterns, "rgraph", achieves a similar level of efficiency to the pure monotonic search, "mon". The computational cost of the "m5j3" search is clearly higher than the cost of the "rgraph" search, despite both algorithms being of the same complexity. The "m5j3" search graph contains about three times more partial hypotheses (thus arcs) than the corresponding "rgraph" search graph.

5 Conclusions

We have shown how translation accuracy can be improved by means of coupling reordering and decoding at a very low computational cost.

The use of linguistically motivated reordering patterns to harmonize the source and target word order has been proved to be an efficient reordering approach. Additionally, the use of source-reordered translation units produces an interesting way to model reordering by means of the n-gram translation model and also reduces the data sparseness problem of the translation model caused by using longer tuples.

We have also seen that coupling reordering and decoding alleviates the problem of using too general rewrite rules, as reordering decisions are not made solely by the rewrite rules but during the global search between the whole SMT models.

Efficiency results have shown a significant reduction in search space (and thus in time) when comparing the reordering approach presented to a brute-force approach. It achieves similar results to a pure monotonic search.

So far, we have carried out experiments applying the reordering framework on different language pairs, such as Chinese–English and Arabic–English, achieving interesting results in terms of efficiency, although no accuracy improvements have yet been reached. Further work must be undertaken towards overcoming the difficulty shown by the approach when dealing with long-distance patterns.

Acknowledgements This work has been partially funded by the European Union under the integrated project TC-STAR (IST-2002-FP6-506738, http://www.tc-star.org), the Spanish Government under grant TEC2006-13964-C03 (AVIVAVOZ project), and the Universitat Politècnica de Catalunya under UPC-RECERCA grant.

References

Berger AL, Della Pietra SA, Della Pietra VJ (1996) A maximum entropy approach to natural language processing. Comput Ling 22:39–72

Brants T (2000) TnT—A statistical part-of-speech tagger. In: Association for Computational Linguistics 6th applied natural language processing conference, Seattle, Washington, pp 224–231

Brown PE, Cocke J, Della Pietra SA, Della Pietra VJ, Jelinek F, Lafferty JD, Mercer RL, Roossin PS (1990) A statistical approach to machine translation. Comput Ling 16:79–85; repr. in: Nirenburg S, Somers H, Wilks Y (eds) (2003) Readings in machine translation. MIT Press, Cambridge MA, 355–362

Brown PE, Della Pietra VJ, Della Pietra SA, Mercer RL (1993) The mathematics of statistical machine translation: parameter estimation. Comput Ling 19:263–311

Carreras X, Chao I, Padró L, Padró M (2004) FreeLing: an open-source suite of language analyzers. In: 4th international conference on language resources and evaluation, Lisbon, Portugal, pp 239–242

Chiang D (2005) A hierarchical phrase-based model for statistical machine translation. In: ACL-05, 43rd annual meeting of the Association for Computational Linguistics, University of Michigan, pp 263–270

Collins M, Koehn P, Kučerová I (2005) Clause restructuring for statistical machine translation. In: ACL-05, 43rd annual meeting of the Association for Computational Linguistics, University of Michigan, pp 531–540

Costa-jussà MR, Fonollosa JAR (2006) Statistical machine reordering. In: Proceedings of the conference on empirical methods in natural language processing (EMNLP-06), Sydney, Australia, pp 70–76

Crego JM (2005) MARIE: Ngra(m)-based statistical m(a)chine t(r)anslat(i)on d(e)coder, http://gps-tsc.upc.es/veu/soft/soft/marie/, accessed April 5, 2007

Crego JM, Costa-jussà MR, Mariño JB, Fonollosa JAR (2005a) Ngram-based versus phrase-based statistical machine translation. In: International workshop on spoken language translation: Evaluation campaign on spoken language translation, Pittsburgh, PA

Crego JM, Mariño JB (2006a) Integration of POStag-based source reordering into SMT decoding by an extended search graph. In: AMTA 2006, Proceedings of the 7th conference of the Association for Machine Translation in the America, Visions for the future of machine translation, Cambridge, MA, pp 29–36

Crego JM, Mariño JB (2006b) Reordering experiments for N-gram-based SMT. In: IEEE/ACL 2006 Workshop on spoken language technology, Palm Beach, Aruba

Crego JM, Mariño JB, de Gispert A (2004) Finite-state-based and phrase-based statistical machine translation. In: INTERSPEECH 2004–ICSLP, 8th international conference on spoken language processing, Jeju Island, Korea, pp 37–40

Crego JM, Mariño JB, de Gispert A (2005b) An n-gram-based statistical machine translation decoder. In: Interspeech'2005–Eurospeech, 9th European conference on speech communication and technology Lisbon, Portugal, pp 3185–3188

Crego JM, Mariño JB, de Gispert A (2005c) Reordered search and tuple unfolding for Ngram-based SMT. In: The tenth machine translation summit, Phuket, Thailand, pp 283–289

Doddington G (2002) Automatic evaluation of machine translation quality using *n*-gram co-occurrence statistics. In: ARPA workshop on human language technology notebook proceedings, San Diego, CA, pp 139–145

Kanthak S, Vilar D, Matusov E, Zens R, Ney H (2005) Novel reordering approaches in phrase-based statistical machine translation. In: ACL-05 workshop, Building and using parallel texts: Data-driven machine translation and beyond, Ann Arbor, Michigan, pp 167–174

Knight K (1999) Decoding complexity in word replacement translation models. Comput Ling 26:607–615

Koehn P, Axelrod A, Birch A, Callison-Burch C, Osborne M, Talbot D (2005) Edinburgh system description for the 2005 IWSLT speech translation evaluation. In: International workshop on spoken language translation: Evaluation campaign on spoken language translation, Pittsburgh, PA

Kumar S, Byrne W (2005) Local phrase reordering models for statistical machine translation. In: HLT/EMNLP 2005 human language technology conference and conference on empirical methods in natural language processing, Vancouver, British Columbia, pp 161–168

Langlais P, Gotti F (2005) Phrase-based SMT with shallow tree-phrases. In: HLT-NAACL 06 statistical machine translation workshop, New York City, pp 39–46

Mariño JB, Banchs RE, Crego JM, de Gispert A, Lambert P, Fonollosa JAR, Costa-jussà MR (2006) N-gram-based machine translation. Comput Ling 32:527–549

Nagata M, Saito K, Yamamoto K, Ohashi K (2006) Clustered global phrase reordering model for statistical machine translation. In: COLING·ACL 2006, 21st international conference on computational linguistics and 44th annual meeting of the Association for Computational Linguistics, Sydney, Australia, pp 713–720

Nelder J, Mead R (1965) A simplex method for function minimization. Comput J 7:308–313

Och FJ (2003) Minimum error rate training for statistical machine translation. In: 41st annual meeting of the Association for Computational Linguistics, Sapporo, Japan, pp 160–167

Och FJ, Ney H (2000) Improved statistical alignment models. In: 38th annual meeting of the Association for Computational Linguistics, Hong Kong, China, pp 440–447

Och FJ, Ney H (2002) Discriminative training and maximum entropy models for statistical machine translation. In: 40th annual meeting of the Association for Computational Linguistics, Philadelphia, PA, pp 295–302

Papineni K, Roukos S, Ward T, Zhu W-J (2002) Bleu: A method for automatic evaluation of machine translation. In: 40th annual meeting of the Association for Computational Linguistics, Philadelphia, PA, pp 311–318

Popovic M, Ney H (2006) POS-based word reorderings for statistical machine translation. In: LREC-2006: Fifth international conference on language resources and evaluation, Genova, Italy, pp 1278–1283

Quirk C, Menezes A, Cherry C (2005) Dependency treelet translation: Syntactically informed phrasal SMT. In: ACL-05, 43rd annual meeting of the Association for Computational Linguistics, University of Michigan, pp 271–279

Simard M, Cancedda N, Cavestro B, Dymetman M, Gaussier E, Goutte C, Yamada K, Langlais P, Mauser A (2005) Translating with non-contiguous phrases. In: HLT/EMNLP 2005 human language technology conference and conference on empirical methods in natural language processing, Vancouver, British Columbia, pp 755–762

Stolcke A (2002) SRILM—an extensible language modeling toolkit. In: 7th international conference on spoken language processing, Denver, CO, pp 901–904

Tillmann C (2004) A unigram orientation model for statistical machine translation. In: HLT-NAACL 2004, Human language technology conference and North American chapter of the Association for Computational Linguistics annual meeting, Short papers, Boston, USA, pp 101-104

Tillmann C, Zhang T (2005) A localized prediction model for statistical machine translation. In: ACL-05, 43rd annual meeting of the Association for Computational Linguistics, University of Michigan, pp 557–564

Watanabe T, Tsukada H, Isozaki H (2006) Left-to-right target generation for hierarchical phrase-based translation. In: COLING·ACL 2006, 21st international conference on computational linguistics and 44th annual meeting of the Association for Computational Linguistics, Sydney, Australia, pp 777–784

Wu D (1996) A polynomial-time algorithm for statistical machine translation. In: 34th annual meeting of the Association for Computational Linguistics, Santa Cruz, CA, pp 152–158

Wu D (1997) Stochastic inversion transduction grammars and bilingual parsing of parallel corpora. Comput Ling 23:377–404

Xia F, McCord M (2004) Improving a statistical MT system with automatically learned rewrite patterns. In: 20th international conference on computational linguistics, Geneva, Switzerland, pp 508–514

Yamada K, Knight K (2002) A decoder for syntax-based statistical MT. In: 40th annual meeting of the Association for Computational Linguistics, Philadelphia, PA, pp 303–310

Zens R, Ney H (2006) Discriminative reordering models for statistical machine translation. In: HLT-NAACL 06 statistical machine translation workshop, New York City, pp 55–63

Zens R, Och FJ, Ney H (2002) Phrase-based statistical machine translation. In: Jarke M, Koehler J, Lakemeyer G (eds) KI 2002: Advances in artificial intelligence, 25th annual German conference on AI, KI 2002, Aachen, Germany. Springer Verlag, Berlin, Germany, pp 191–198