Reordered Search and Tuple Unfolding for Ngram-based SMT

Josep M. Crego, José B. Mariño and Adrià de Gispert
TALP Research Center
Signal Theory and Communications Department
Universitat Politècnica de Catalunya, Barcelona
{jmcrego,canton,agispert}@gps.tsc.upc.edu

Abstract

In Statistical Machine Translation, the use of reordering for certain language pairs can produce a significant improvement on translation accuracy. However, the search problem is shown to be NP-hard when arbitrary reorderings are allowed. This paper addresses the question of reordering for an Ngram-based SMT approach following two complementary strategies, namely reordered search and tuple unfolding. These strategies interact to improve translation quality in a Chinese to English task.

On the one hand, we allow for an Ngram-based decoder (MARIE) to perform a reordered search over the source sentence, while combining a translation tuples Ngram model, a target language model, a word penalty and a word distance model. Interestingly, even though the translation units are learnt sequentially, its reordered search produces an improved translation.

On the other hand, we allow for a modification of the translation units that unfolds the tuples, so that shorter units are learnt from a new parallel corpus, where the source sentences are reordered according to the target language. This tuple unfolding technique reduces data sparseness and, when combined with the reordered search, further boosts translation performance.

Translation accuracy and efficency results are reported for the IWSLT 2004 Chinese to English task.

1 Introduction

It is widely known that one of the hardest difficulties that statistical machine translation (SMT) systems face is reordering. This linguistic phenomenon is especially relevant in a task with two languages belonging to different families that tend to have different syntactic structures (such as Chinese and English).

Usually, reorderings are divided into short-distance (or local) and long-distance reorderings. Whereas the former are generally well captured by the use of phrases as translation units, long-distance reorderings require SMT decoders to allow for discontinuities when translating a given source sentence. This means that the target sentence is generated by translating parts of the source sentence in a non-sequential fashion. However, this search can cause a combinatory explosion of the search graph if arbitrary reorderings are allowed, being an NP-hard problem (Knight, 1999).

This issue has been tackled in previous work for a phrase-based SMT approach, typically reducing the combinatory explosion of the search space by using a distortion model that penalizes the longest reorderings, which are only allowed if well supported by the other feature models involved in the search (Och and Ney, H., 2004), (Koehn, 2004).

In this paper we deal with the question of reordering for an Ngram-based SMT approach with two complementary strategies, namely reordered search and tuple unfolding. These strategies interact to improve translation quality in a Chinese to English task.

On the one hand, we introduce reordering capabilities into an Ngram-based decoder. The decoder then performs a reordered search over the source sentence, and combines a translation tuples Ngram model, a target language model, a word penalty and a word distance model. Interestingly, even though the translation units are learnt sequentially, its reordered search produces an improved translation.

Furthermore, we allow for a modification of the translation units that unfolds the tuples, so that shorter units are learnt from a new parallel corpus, where the source sentences are reordered according to the target language. This tuple unfolding reduces data sparseness and, when combined with the reordered search, further boosts translation performance.

The paper is organized as follows. Section 2 reviews the main characteristics of Ngram-based SMT modeling, presenting the models that play a role in the log-linear combination in order to decide for the best translation. Section 3 discusses the decoder used (MARIE), giving details on pruning strategies to constrain the reordered search. The tuple unfolding technique is introduced in section 4, while experiments for the Chinese to English IWSLT 2004 task are reported in section 5. Finally, section 6 concludes and outlines further research.

2 Ngram-based SMT Modeling

Statistical Machine Translation (SMT) is thought as a task where each source sentence f_1^J is transformed into (or generates) a target sentence e_1^I by means of a stochastic process. The translation of a source sentence f_1^J can be formulated as the search of the target sentence e_1^I that maximizes the conditional probability $p(e_1^I|f_1^J)$, which can be rewritten using the Bayes rule as:

$$\arg\max_{e_1^I} \left\{ p(f_1^J|e_1^I) \cdot p(e_1^I) \right\} \tag{1}$$

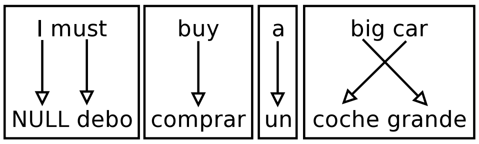
Alternatively to this classical source channel approach, the posterior probability $p(e_1^I|f_1^J)$ can be modeled directly as a log-linear combination of feature models (Och and Ney, H., 2002), based on the maximum entropy framework, as shown in (Berger et al., 1996). This simplifies the introduction of several additional models explaining the translation process, as the search becomes:

$$\arg\max_{e_1^I}\{exp(\sum_i \lambda_i h_i(e,f))\} \tag{2}$$

where the feature functions h_i are the system models (translation model, language model, reordering model, ...), and the λ_i weights are typically optimized to maximize a scoring function on a development set.

In this work a combination of 4 feature models is used, which include:

- a translation Ngram tuple-based model
- a target Ngram language model
- a word penalty
- a word distance-based reordering model



I must # debo
buy # comprar
a # un
big car # coche grande

Figure 1: *Tuples extraction from a pair of word aligned source sentences.*

2.1 Translation model

As for the Translation Model, it is defined as a language model of the parallel corpus, expressed in bilingual units (here called tuples). Tuples are extracted from a word-to-word aligned corpus. More specifically, word-to-word alignments are performed in both directions, source-to-target and target-to-source, by using GIZA++ (Och and Ney, H., 2000). Then, tuples are extracted from the union set of alignments according to the following constraints (Crego et al., 2004):

- a monotonous segmentation of each bilingual sentence pairs is produced,
- no word inside the tuple is aligned to words outside the tuple, and
- no smaller tuples can be extracted without violating the previous constraints.

As a consequence of these constraints, only one segmentation is possible for a given sentence pair. Figure 1 presents a simple example illustrating the tuple extraction process.

This Translation Model is implemented using an Ngram language model (with $N = 3$), as expressed by the following equation:

$$p(e,f) = Pr(t_1^K) = \prod_{k=1}^{K} p(t_k \,|\, t_{k-2}, t_{k-1}) \tag{3}$$

2.2 Other feature models

The other three feature functions (defined as model probabilities) are:

- An Ngram target language model (with $N = 3$):

$$Pr(e_1^I) = \prod_{i=1}^{I} p(e_i \,|\, e_{i-2}, e_{i-1}) \tag{4}$$

- A standard word penalty used to compensate the preference for shorter translations caused by the presence of the target language model:

$$Pr(e_1^I) = exp(I) \tag{5}$$

- and a word distance-based reordering model.

$$Pr(t_1^K) = exp(-\sum_{k=1}^{K} d_k) \tag{6}$$

  where d_k is the distance between the first word of the K^{th} tuple (unit), and the last word +1 of the $K-1^{th}$ tuple (distances are measured in words referring to the units source side). Obviously, this model intervenes only when the decoder performs a reordered search.

The following section introduces the decoder that generates the best translation hypothesis taking these feature models into account.

3 Decoder

In SMT decoding, translated sentences are built incrementally from left to right in form of hypotheses, allowing for discontinuities in the source sentence.

A Beam search algorithm with pruning is used to find the optimal path. The search is performed by building partial translations (hypotheses), which are stored in several lists. These lists are pruned out according to the accumulated probabilities of their hypotheses.

Worst hypotheses with minor probabilities are discarded to make the search feasible.

3.1 Search Graph Structure

Hypotheses are stored in different lists depending on the number of source and target words already covered.

Figure 2 shows an example of the search graph structure. It can be decomposed into three levels:

- Hypotheses. In figure 2, represented using '*'.
- Lists. In figure 2, the boxes with a tag corresponding to its covering vector. Every list contains an ordered set of hypotheses (all the hypotheses in a list have translated the same words of the source sentence).
- Groups (of lists). In figure 2, delimited using dotted lines. Every group contains an ordered set of lists, corresponding to the lists of hypotheses covering the same number of source words (to order the lists in one group the cost of their best hypothesis is used). When the search is restricted to monotonous translations, only one list is allowed on each group of lists.

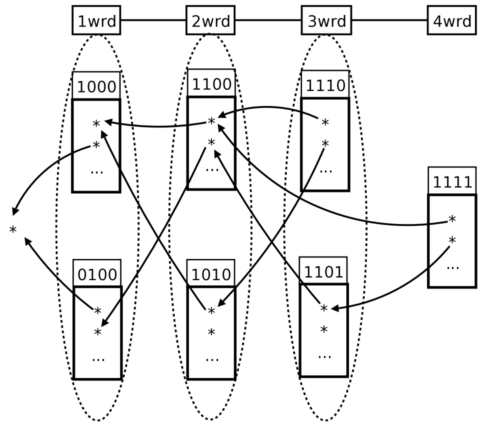


Figure 2: *Search graph corresponding to a source sentence with four words. Details of constraints are given in following sections.*

The search loops expanding available hypotheses. The expansion proceeds incrementally starting in the group of lists covering 1 source word, ending with the group of lists covering $J-1$ source words (J is the size in words of the source sentence).

See (Crego et al., 2005) for further details.

3.2 Pruning Hypotheses

The search graph structure is thought to perform very accurate comparisons (only hypotheses covering the same source words are compared) in order to allow for very high pruning levels. Despite of this, the number of lists when allowing for reordering grows exponentially (an upper bound is 2^J, where J is the number of words of the source sentence) and forces the search to be further pruned out for efficiency reasons.

Only the best N hypotheses are kept on each list (histogram pruning), with best scores within a margin, given the best score in the list (threshold pruning). Not just the lists, but the groups are pruned out, following the same pruning strategies. To score a list, the cost of its best scored hypothesis is used.

3.3 Reordering capabilities

When allowing for reordering, the pruning strategies are not enough to reduce the combinatory explosion without an important lost in translation performance. With this purpose, two reordering strategies are used:

- A distortion limit (m). A source word (phrase or tuple) is only allowed to be reordered if it does not exceed a distortion limit, measured in words.
- A reorderings limit (j). Any translation path is only allowed to perform j reordering jumps.

The use of the reordering strategies suppose a necessary trade-off between quality and efficiency. Further details of these reordering strategies are given in the experiments reported in section 5.

4 Tuple unfolding

4.1 Motivation

A complementary approach to translation with reordering can be followed if we allow for a certain reordering in the training data. This means that the translation units are modified so that they are not forced to sequentially produce the source and target sentences anymore. In this case, if we do not want to lose the information on the correct order, this must be provided otherwise.

One possibility is to modify the translation units changing the order of the target words, encoding the reordering information by means of position indexes while decoding is still monotonous, as was done in (de Gispert and Mariño, 2003). However, results do not seem to be very promising, as translation is highly improved when reordering is able to deal with unseen examples (never seen in train).

On the contrary, here we propose a more general modification, which unfolds the translation units so that the order of the source words is modified according to the order of the target words. The rationale of this approach is double. On the one hand, it makes sense when applied into a decoder with reordering capabilities as the one presented in the previous section, since we need correct-order translation hypotheses to evaluate target language models during the search. And on the other hand, the unfolding process generates shorter tuples, alleviating the problem of embedded units (tuples only appearing within long distance alignments and not having a translation in isolation) (de Gispert and Mariño, 2004), which can be very relevant in a Chinese – English task. Now most of embedded words appear as single units and with a bilingual context defined by the N-gram bilingual model.

4.2 Unfolding technique

The extract-unfold-tuples algorithm is here outlined. It uses the word to word alignments obtained by any alignment procedure. It is decomposed in two steps:

- 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.
- The second step consists on to output the resulting groups (tuples), keeping the word order of target sentece words. Though, the tuples sequence modifies the source sentence word order.

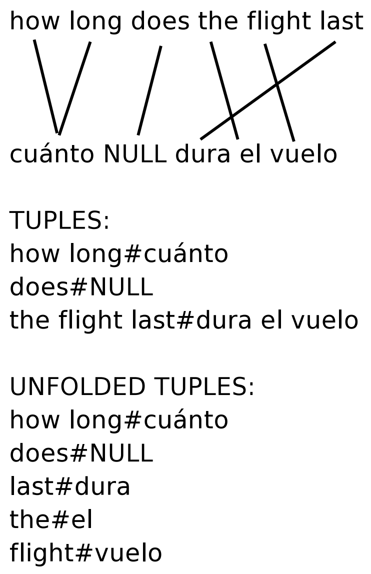


Figure 3: *Different bilingual units (tuples) are extracted using the extract-tuples and extract-unfold-tuples methods. As can be seen,to produce the source sentence, the extracted unfolded tuples must be reordered. It is not the case of the target sentence, as it can be produced in order using both sequence of units.*

Figure 3 shows the bilingual units extracted using the extract-tuples and extract-unfold-tuples methods, for a given word to word aligned sentence pair.

Table 1 shows the main differences of the resulting bilingual units when using the extract-tuples (Crego et al., 2004) method, and when using the extract-unfold-tuples method (outlined in the previuos lines) for the BTEC[1] corpus.

	regular tuples	unfold tuples
total	94.920	106.080
vocab	26.873	30.803
avg size	3.9	3.5

Table 1: *Statistics consist of the number of tuples extracted from the whole corpus, the vocabulary of tuples and the mean size in words of tuples (including source and target sides).*

5 Experiments and results

5.1 Experimental Framework

Experiments have been carried out using the IWSLT 2004 BTEC corpus from Chinese to English.

Table 2 shows the main statistics of the used data, namely number of sentences, words, vocabulary, and average sentence length for each language.

Set	Lng	Sent	Words	Voc	Avg
trn	eng	20,000	188,935	8,191	9.4
	chi		182,904	7,643	9.1
dev	chi	506	3,515	870	6.9
tst	chi	500	3,794	893	7.6

Table 2: *BTEC Corpus. For the English test and dev sides of the corpus, 16 different references are available.*

We used GIZA++ to perform the word alignment of the whole training corpus, and refined the links by the union of both alignment directions (Och and Ney, H., 2004). Afterwards we segmented the bilingual sentence pairs of the training set, extracting translation units (tuples) using the extract-tuples method described in (Crego et al., 2004).

To train the Ngram models, we used the SRILM toolkit (Stolcke, 2002). The type of discounting algorithm used was the modified Kneser-Ney combining higher and lower order estimates via interpolation.

[1]www.slt.atr.jp/IWSLT2004

The weights λ_i for the log-linear combination of models were set in order to minimize the BLEU score (Papineni et al., 2001) on the development set, using the simplex (Nelder and Mead, 1965) algorithm.

All the experiments were performed on a Pentium IV (Xeon 3.06GHz), with 4Gb of RAM memory.

5.2 Experiments

The following subsections introduce the experiments that have been carried out in order to evaluate the presented techniques, discussing the results obtained.

The evaluation has been carried out using references and translations in lowercase and without punctuation marks.

5.2.1 Monotonous vs. reordered search

Results when decoding monotonously and with reordering are compared, both in terms of translation quality and efficiency, which turns out to be very important in the case of reordering.

When allowing for reordering, the search space is pruned out using the reordering strategies explained in section 3. For the experiments reported in this paper, the distortion limit (m) and reorderings limit (j) have been empirically set to 5 and 3, as they showed a good trade-off between quality and efficiency.

5.2.2 Regular vs. unfolded tuples

Results when training the translation model with regular tuples (without modification in order) and with unfolded tuples are also compared.

5.3 Results

Config.	BLEU	mWER	mPER	time
baseline	28.85	53.42	42.89	-
tpl.mon	33.14	51.5	41.53	11
tpl.reo	36.33	49.68	41.41	333
utpl.mon	33.16	50.16	41.25	12
utpl.reo	37.82	47.31	39.9	354

Table 3: *Translation quality (using BLEU, mWER and mPER) and efficiency (measured in decoding time) results. The first row shows the results of the baseline TALP system.*

Table 3 shows the results in BLEU, mWER and mPER, as well as the time (in seconds) needed to decode, obtained for the following experiments:

- monotonous decoding and regular tuples (**tpl.mon**)
- reordered decoding and regular tuples (**tpl.reo**). With distortion limit set to 5 and reorderings limit set to 3.
- monotonous decoding and unfolded tuples (**utpl.mon**)
- reordered decoding and unfolded tuples (**utpl.reo**). With distortion limit set to 5 and reorderings limit set to 3.

Additionally, the results of the baseline TALP system (with monotonous decoding, regular tuples and without feature models besides the translation model) obtained in the evaluation campaign of the International Workshop on Spoken Language Translation (IWSLT) in 2004 are presented (**baseline**).

The algorithms used for computing all the evaluation measurements (mWER, mPER and BLEU) were the official TC-STAR evaluation tools distributed by ELDA (`http://www.elda.org/`), wich were not the same than those used for the same measures in the IWSLT 2004 evaluation campaign (slight differences can be observed in both evaluations).

5.4 Discussion

First of all, we observe a pronounced improvement of translation quality by the use of additional feature models (target LM and word penalty) and the MARIE decoder in the monotone configuration (**tpl.mon**) with respect to the evaluation with the TALP decoder presented in IWSLT 2004 for the same data.

Regarding reordering, the MARIE decoder achieves a clear improvement in performance at a clear efficiency cost (see **tpl.reo**). Interestingly, this improvement is only significant in BLEU and mWER, whereas mPER stays constant.

To our view, it shows how the reordered search helps to improve the order of the translated sentences, but it cannot modify the decision as to which translation each tuple has (these are learnt sequentially with many embedded units).

However, when translating from the model learnt from unfolded tuples (**utpl.reo**), reordering does provide a further boost, both in BLEU and mWER, *and* in mPER. The effect of the reordered search and the unfolded tuples add up to produce a better output.

On the one hand, the search is still helping the decoder to produce a sentence with better correct order *and* at the same time the translation units are better selected.

The BLEU score reflects this double effect with an increase to 37.82 in the case of reordered search, where only 36.33 was achieved with regular tuples.

Finally, the monotone decoding with unfolded tuples (**utpl.mon**) is very similar to the case with regular tuples, although it seems to achieve a better ordering, as shown in the decrease in mWER from 51.5 to 50.16. Old embedded units that cannot be used in isolation and which are now revealed by the unfolding technique seem to account for this improvement.

6 Conclusions and Further Work

In this paper we have addressed the reordering problem for an Ngram-based SMT system through performing a constrained reordered search, and unfolding the bilingual units extracted from the corpus in the training process.

Results have shown the suitability of the presented strategies to improve the translation quality.

The use of the reordering search strategies has allowed to efficiently perform a reordered search at a low cost in translation quality. Further experiments have been performed relaxing the reordering constraints achieving a slight improvement at a very high cost in decoding time.

In general, the use of reordering helps the decoder to choose the right word order of the translated sentences, as shows the mWER reduction when reordering is applied.

When extracting bilingual units, the change of order performed in the sentence source words has allowed to improve the modeling of the translation units (shorter units implies a data sparseness reduction), as shows the mPER reduction when the extract-unfold-tuples method is used to extract units.

In the future we would like to explore more sophisticated reordering search strategies in order to improve the ability of the decoder to discard the worst reordering alternatives. We are also investigating the use of a heuristic function to better prune out the search space.

7 Acknowledgements

This work has been partially supported by the Spanish government, under grant TIC-2002-04447-C02 (Aliado Project), the European Union, under FP6-506738 grant (TC-STAR project) and the Universitat Politècnica de Catalunya (UPC), under UPC-RECERCA grant.

References

A. Berger, S. Della Pietra, and V. Della Pietra. 1996. A maximum entropy approach to natural language processing. *Computational Linguistics*, 22(1):39–72, March.

J.M. Crego, J. Mariño, and A. de Gispert. 2004. Finite-state-based and phrase-based statistical machine translation. *Proc. of the 8th Int. Conf. on Spoken Language Processing, ICSLP'04*, pages 37–40, October.

J.M. Crego, J. Mariño, and A. de Gispert. 2005. An ngram-based statistical machine translation decoder. *submitted to ICSLP05.*

A. de Gispert and J. Mariño. 2003. Experiments in word-ordering and morphological preprocessing for transducer-based statistical machine translation. *IEEE Automatic Speech Recognition and Understanding Workhsop, ASRU'03*, pages 634–639, November.

A. de Gispert and J. Mariño. 2004. Talp: Xgram-based spoken language translation system. *Proc. of the Int. Workshop on Spoken Language Translation, IWSLT'04*, pages 85–90, October.

K. Knight. 1999. A statistical machine translation tutorial workbook. http://www.clsp.jhu.edu/ws99/projects /mt/wkbk.rtf, April.

P. Koehn. 2004. Pharaoh: a beam search decoder for phrase-based statistical machine translation models. *Proc. of the 6th Conf. of the Association for Machine Translation in the Americas*, pages 115–124, October.

J.A. Nelder and R. Mead. 1965. A simplex method for function minimization. *The Computer Journal*, 7:308–313.

F.J. Och and Ney, H. 2000. Improved statistical alignment models. *38th Annual Meeting of the Association for Computational Linguistics*, pages 440–447, October.

F.J. Och and Ney, H. 2002. Discriminative training and maximum entropy models for statistical machine translation. *40th Annual Meeting of the Association for Computational Linguistics*, pages 295–302, July.

F.J. Och and Ney, H. 2004. The alignment template approach to statistical machine translation. *Computational Linguistics*, 30(4):417–449, December.

K. Papineni, S. Roukos, T. Ward, and W. Zhu. 2001. Bleu: a method for automatic evaluation of machine translation. Technical Report RC22176 (W0109-022), IBM Research Division, Thomas J. Watson Research Center.

A. Stolcke. 2002. Srilm - an extensible language modeling toolkit. *Proc. of the 7th Int. Conf. on Spoken Language Processing, ICSLP'02*, September.