Using Reordering in Statistical Machine Translation based on Recursive Alignment Block Classification

Marta R. Costa-jussà and José A. R. Fonollosa and Enric Monte

TALP Research Center
{mruiz,adrian,enric}@gps.tsc.upc.edu

Abstract

Statistical Machine Translation (SMT) is based on alignment models which learn from bilingual corpora the word correspondences between the source and target language. These models are assumed to learn word reorderings. However, the difference in word order between two languages is one of the most important sources of errors in SMT. This paper proposes a Recursive Alignment Block Classification algorithm (RABCA) that can take advantage of inductive learning in order to solve reordering problems. This algorithm should be able to cope with swapping examples seen during training; it should infer properties that might permit to reorder pairs of blocks (sequences of words) which did not appear during training; and finally it should be robust with respect to training errors and ambiguities. Experiments are reported on the EuroParl task and RABCA is tested using two state-of-the-art SMT systems: a phrased-based and an Ngram-based. In both cases, RABCA improves the translation quality.

1. Introduction

The introduction of Statistical Machine Translation (SMT) has yielded significant improvements over the initial word-based translation. At the end of the last decade the use of context in the translation model (phrase-based approach) represented a clear improvement in the translation quality (Zens et al., 2004).

In parallel to the phrase-based approach, the use of a language model of bilingual units gives comparable results to the phrase-based approach (Mariño et al., 2006).

In both systems, the introduction of some reordering capabilities is of crucial importance for some language pairs.

In our approach, we introduce order modifications to the source corpora so that alignments and translation become more monotonic by using the alignment information and a classification algorithm. The proposed algorithm parses alignments to detect reorderings. Unseen candidates to reordering are dealt with pairs of swapping blocks (sequences of words) belonging to an specific group. These block groups have been created following recursively a co-occurrence block criterion.

The report is organized as follows. Section 2 reports the reordering process and the algorithm which infers the reorderings. Section 3 briefly describes the two baseline systems: phrase-based and Ngram-based system. Both capable of producing state-of-the-art SMT translations. Section 4 sets the Evaluation Framework and dicusses the experiments and results. Finally, Section 5 draws the conclusions.

2. Reordering based on Alignment Blocks Classification

2.1. Motivation

SMT systems are trained by using a bilingual corpus composed of bilingual sentences. Each bilingual sentence is composed of a source and target sentence, and we align them at the word level by using GIZA++ (Och and Ney, 2003). Generally, this alignment contains a certain amount of errors which deteriorates the translation quality. One way of improving this alignment is by monotonization (Kanthak et al., 2005), i.e. reordering the words in the source sentence following the order of the words in the target sentence. For instance in a Spanish to English translation, the original sentence *El discurso político fue largo* would be modified as *El político discurso fue largo*. And it would monotonize the alignment: *El#The político#political discurso#speech fue#was largo#long*. In (Popovic and Ney, 2006), they perform rules based on Part of Speech (POS) tags and reorder pairs of words both in the training and test sentences. Similarly, we propose one type of monotonization: pairs of consecutive blocks which swap if swapped generate a correct monotonic translation. The main difference with (Popovic and Ney, 2006) is that our approach learns the blocks which swap instead of following a pre-defined set of rules. Figure 1 shows an example of this type of pairs. The reordering based in blocks covers most cases as shown in (Tillmann and Zhang, 2005).

2.2. Reordering process

Our purpose is to model the effect of local block reordering to: (1) monotonize the source training corpus; and (2) generalize this monotonization in the test stage to perform a correct monotonic translation. In order to fulfil (1) and (2), the reordering process consists of the following steps:

- Given a word alignment, we extract a List of Alignment Blocks (LAB). An Alignment Block consists of a pair of consecutive source blocks whose target translation is swapped. See Figure 1.
- Given the LAB, we apply the Recursive Alignment Block Classification (RABCA), see section 2.3., which allows us to decide whether two consecutive blocks have to be reordered or not.
- We use the criteria of the RABCA to reorder the source corpora (including training, development and test sets).
- Given the monotonized source training corpus, we realign it with the original target training corpus.

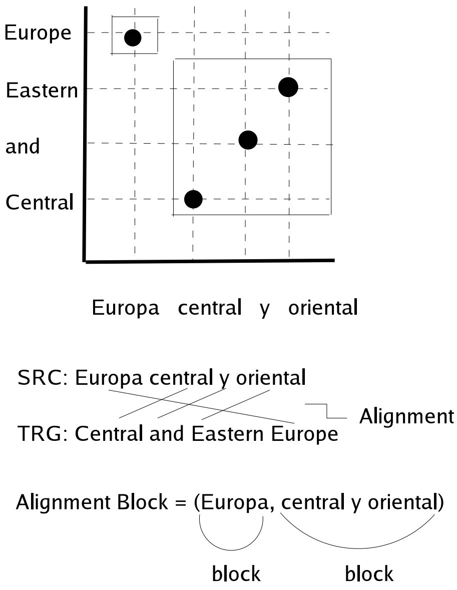
Europe
Eastern
and
Central
Europa central y oriental
SRC: Europa central y oriental
TRG: Central and Eastern Europe
Alignment
Alignment Block = (Europa, central y oriental)
block
block

Figure 1: *Example of an Alignment Block, i.e. a pair of consecutive blocks whose target translation is swapped*

- Given the monotonized alignment, we build the SMT systems and translate the monotonized source test set.

2.3. Recursive Alignment Block Classification Algorithm

The objective of this algorithm is to perform block reorderings in case the order of the blocks swaps from source to target. These block reorderings will take into account reorderings that appear in the database, and will generalize to unseen blocks. The set of blocks to be swapped will be denoted as Generalization groups (Gg). The algorithm should be able to cope with swapping examples seen during training; it should infer properties that might allow reordering in pairs of blocks not seen together during training; and finally it should be robust with respect to training errors and ambiguities.
The algorithm consists of two steps:

1. Given the LAB, the algorithm filters the ambiguous Alignment Blocks (i.e. either misaligned or inherently ambiguous). We will define the filtered LAB as LAB_{filt}, which will be a subset of LAB and consists of m pairs of blocks $\{(\alpha_1, \beta_1), (\alpha_2, \beta_2), \dots, (\alpha_m, \beta_m)\}$.

2. From the LAB_{filt}, we create the sets $A = \{\alpha_1, \alpha_2, \dots, \alpha_m\}$ and $B = \{\beta_1, \beta_2, \dots, \beta_m\}$ and the groups $G_1 \dots G_n \dots G_N$. A given group G_n is created following recursively a co-occurrence block criterion (see 2.4.) and has the form $G_n = \{(\alpha_1, \beta_1), \dots (\alpha_p, \beta_p)\}$ where p is the cardinality of G_n; within each group G_n we create also the sets $A_n = \{\alpha_1, \dots \alpha_p\}$ and $B_n = \{\beta_1, \dots \beta_p\}$. From each group G_n we build a Generalization group (Gg_n, where $n = 1, \dots N$) defined as the Cartesian product between the subsets $A_n \in A$ and $B_n \in B$, i.e. $Gg_n = A_n \times B_n$ which will allow to reorder cases such as (α_r, β_s) with $\alpha_r \in A_n$, $\beta_s \in B_n$ and $(\alpha_r, \beta_s) \notin LAB_{filt}$. We can deal with possible inconsistencies, by increasing the filtering threshold, and therefore limiting the number of allowed unseen pairs, and also by processing with morphological information. Note that we assume that only the elements in Gg_n that appear in the training and test corpus set are correct generalizations (see Subsection 4.2).

2.4. Outline of the algorithm

The first phase of the algorithm filters the possible bad alignments or ambiguities, by using the following criteria:

- Pairs appearing less than N_{min} times are discarded.

- Pairs of blocks with a swapping probability (P_{swap}) less than a threshold are also discarded. We define the swapping probability as the ratio between the number of times that two blocks are swapped and the total number of times that the same two blocks appear consecutively.

The second phase of the algorithm infers the generalization groups Gg_n from the filtered LAB (LAB_{filt}).

- Given the LAB_{filt}, the generalization groups Gg_n are constructed as follows:

  1. Initialization: set $n \leftarrow 1$ and $LAB_{tmp} \leftarrow LAB_{filt}$.
  2. Main part: while LAB_{tmp} is not empty do
     - $G_n = \{(\alpha_k, \beta_k)\}$ where (α_k, β_k) is any element of LAB_{tmp}
     - Recursively, move elements (α_i, β_i) from LAB_{tmp} to G_n if there is an element $(\alpha_j, \beta_j) \in G_n$ such that $\alpha_i = \alpha_j$ or $\beta_i = \beta_j$
     - Increase n (i.e. $n \leftarrow n + 1$)
  3. Ending: For each G_n, construct the two sets A_n and B_n which consists of the first and second element of the pairs in G_n, respectively. Then the Cartesian product of A_n and B_n is assigned to Gg_n, i.e. $Gg_n \leftarrow A_n \times B_n$.

2.5. Using extra information

Additionally, the Alignment Block Classification can be used for extracting blocks from a lemmatized corpora. The resulting Gg_n will be able to deal with grammar agreement between elements of each block, for instance the pair $(conferencia, parlamentario)$, which does not have gender agreement, would be a correct generalization if we take each block as a lemma. And, therefore the generalization would not be influenced by the particular distribution of word inflexions in training database.
Furthermore, we can use a tagger to find out the grammatical function of each word. In case the blocks are constituted of only one word, i. e. the Alignment Blocks are pairs of swapping words, a general grammar rule to take into account for the Spanish to English translation is that in Spanish most adjectives are placed after the noun, whereas in English it is the opposite. However, there are exceptions to this rule and we can not rely completely on it.

The use of this morphological information is optional. The algorithm itself does not require extra information that is not employed in an standard SMT system. But it is interesting to benefit from morphological information, if available, in SMT as have been shown in other studies (Nießen and Ney, 2004; de Gispert, 2005).

3. Baseline systems

Two baseline systems are proposed to test our approach. The main difference between the two systems is the translation model, which constitutes the actual core of the translation systems. In both cases it is based on bilingual units. A bilingual unit consists of two monolingual fragments, where each one is assumed to be the translation of the other.

3.1. N-gram Translation Model

The translation model can be thought of a Language Model of bilingual units (here called tuples). These tuples define a monotonic segmentation of the training sentence pairs (f_1^J, e_1^I), into K units $(t_1, ..., t_K)$.

The translation model is implemented using an Ngram language model, (for $N = 3$):

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

Bilingual units (tuples) are extracted from any word alignment according to the following constraints:

- a monotonic segmentation of each bilingual sentence pairs is produced,
- no word inside a 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. See (Mariño et al., 2006) for further details.

3.2. Phrase-based Translation Model

The basic idea of phrase-based translation is to segment the given source sentence into units (here called phrases), then translate each phrase and finally compose the target sentence from these phrase translations (Zens et al., 2004).

Given a sentence pair and a corresponding word alignment, a phrase (or bilingual phrase) is any pair of m source words and n target words that satisfies two basic constraints:

1. Words are consecutive along both sides of the bilingual phrase,
2. No word on either side of the phrase is aligned to a word out of the phrase.

We limit the maximum size of any given phrase to 7. The huge increase in computational and storage cost of including longer phrases does not provide a significant improvement in quality (Koehn et al., 2003) as the probability of reappearance of larger phrases decreases.

Given the collected phrase pairs, we estimate the phrase translation probability distribution by relative frequency in both directions.

$$P(f|e) = \frac{N(f,e)}{N(e)}$$

$$P(e|f) = \frac{N(f,e)}{N(f)}$$

where N(f,e) means the number of times the phrase *f* is translated by *e*.

3.3. Additional feature functions

In each system, the translation model is combined in a log-linear framework with additional feature functions.

- The target language model consists of an n-gram model, in which the probability of a translation hypothesis is approximated by the product of word n-gram probabilities. As default language model, a standard word-based 5-gram language model is generated with smoothing Kneser-Ney and interpolation of higher and lower order ngrams with the SRILM tool (Stolcke, 2002).
- The forward and backwards lexicon models provide lexicon translation probabilities for each phrase/tuple based on the word IBM model 1 probabilities. For computing the forward lexicon model, IBM model 1 probabilities from GIZA++ source-to-target alignments are used. In the case of the backwards lexicon model, target-to-source alignments are used instead.
- The word bonus model introduces a sentence length bonus in order to compensate the system preference for short output sentences.
- The phrase bonus model introduces a constant bonus per produced phrase and it is only used for the phrase-based system.

All these feature functions are combined in the decoder. The different weights are optimized on the development set applying the *Simplex* algorithm (Nelder and Mead, 1965).

4. Evaluation framework

4.1. Corpus statistics

Experiments have been carried out using the EPPS database (Spanish-English).

The EPPS data set corresponds to the parliamentary session transcriptions of the European Parliament and it is currently available at the Parliament's website [1].

In the case of the results presented here, we have used the version of the EPPS data that was made available by RWTH Aachen University through the TC-STAR consortium[2].

[1] http://www.europarl.eu.int/

[2] TC-STAR (Technology and Corpora for Speech to Speech Translation) is an European Community project funded by the *Sixth Framework Programme*. More information can be found at the consortium website: `http: //www.tc-star.org/`

EPPS	*Spanish*	*English*
Training Sentences	1.3 M	
Words	36.6 M	35 M
Vocabulary	153.1 k	106.5 k
Lemma’s vocabulary	78.3 k	91 k
Development Sentences	430	
Words	15.3 k	16 k
Vocabulary	3.2 k	2.7 k
Lemma’s vocabulary	2.1 k	2.2 k
Test Sentences	840	
Words	22.7 k	20.3 k
Vocabulary	4 k	4.3 k
Lemma’s vocabulary	2.6 k	3.3 k

Table 1: *EuroParl Corpus. The Development data set and the Test data set have 2 references.*

Reord. pattern SPA->ENG	Counts	%
(0,1)(1,0)	392	38.4%
(0,2)(1,0)(2,1)	113	11%
(0,1)(1,2)(2,0)	112	11%
Most freq. patterns	617	60.4%

Table 2: *Reordering patterns for Es2En reference alignment of 500 sentences*

English POS-tagging was carried out using the freely available TNT tagger (Brants, 2000) and lemmatization using $wnmorph$ included in *WordNet* package (Miller et al., 1991).

In Spanish, we used the $FreeLing$ (Carreras et al., 2004) analysis tool which generates the POS-tagging and the lemma for each input word.

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

4.2. Experiments and results

Tunning RABCA parameters: LAB filtering parameters N_{min} and P_{swap} We study most common reordering patterns found in our task. We have a reference corpus which consists of 500 bilingual sentences manually aligned (Lambert et al., 2006). Given the word alignment reference, we extract the reordering patterns. Most common reordering patterns have been described as in (de Gispert and Mariño, 2003): $(x_1, y_1)(x_2, y_2)...(x_N, y_N)$ where each (x_i, y_i) describes a link between position x_i and y_i, in the original and the reordered source sentence composed of the source words appearing in the monotonization of the alignment. This means that the cross (0,1)(1,0) would reflect: a_n b_n to b_n a_n, where a_n (b_n) is only one word. Table 2 presents the most frequent reordering patterns when aligning from Spanish to English with the EuroParl task. Next experiment deals with the most frequent reordering pattern: (0,1)(1,0).

The amount of admissible blocks in the LAB, is a function of the parameters N_{min} and P_{swap} (see Subsection 2.3.). We determine these parameters, from a subset of the corpus as follows. We remove the 500 manually aligned sentences from the training corpus. We train the Alignment Classification Block and swap the reference source set. Given a swapping of two words, it can be a $Success$ (S) if the reference alignment is swapped, or a $Failure$ (F) if the reference alignment is not swapped. Combining these two sources of information, we use the Simplex algorithm to minimise the following:

$$Q = -(N_S - N_F)$$

We have chosen the cost function Q as a coherent criterion to optimize the number of successes (N_S) and minimize the number of failures (N_F). The cost function Q has as argument two quantified variables, and its output is a difference between two integers. Therefore the gradient based optimization techniques are not feasible. Note that the underlying problem is a multi-objective optimization, which we have transformed to a simple optimization problem by giving equal importance to the two objectives; i.e. $Successes$ and $-Failures$. For this kind of problems, direct search techniques such as the Simplex algorithm are adequate.

Figure 2 shows the relation between the two objectives, which gives a curve similar to the ROC curve used in detection theory. An increase in the success rate increases the failure rate in ROC, therefore there is a trade-off between two objectives. The solution that we have selected is the intersection of the diagonal with the curve, which corresponds to a trade-off that gives the same weight to both objectives. The maximum Q corresponds to the curve of lemmas reordering plus tags.

Given the optimum values of N_{min} and P_{swap}, we have also studied the number of correct generalizations, i.e. pairs of words which have been swapped correctly and were not seen swapped during training. Almost half of the $Successes$ are generalizations.

A possible criticism to the proposed method is that some generalized pairs have no sense from the syntactical or semantical point of view: for example, *(conferenciante, parlamentaria)*. Our assumption here is that the source corpus is correctly written. Therefore, the non-sense generalized pairs will not appear in the corpus and will end unused.

Figure 2 shows that the LAB filtering parameters that minimize Q are $N_{min} = 5$ and $P_{swap} = 0.33$.

Reordering experiments in the EuroParl Es2En task Before applying the algorithm, we added morphological information: lemmas and tags. We added the two informations sequentially: firstly, we used the lemma alignment to build the LAB and secondly, we removed from the list those pairs of blocks which were not constituted by noun plus adjective. The latter is done because, in general, *noun* plus *adjective* in Spanish becomes *adjective* plus *noun* in English. There are a few common exceptions as *gran hombre* ⟶ *big man*. That is why we avoid using this rule directly. Afterwards we built the RABCA and finished the reordering process.

Table 3 shows the improvement in both measures mWER and BLEU.

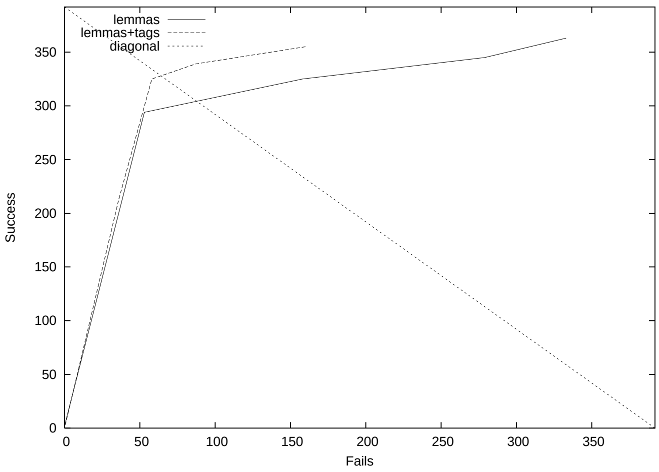

Figure 2: *Relation between number of successes and fails (with different parameters -N_{min}, P_{swap}- for the reordering based on Alignment Block Classification: (1) using lemmas and (2) using lemmas plus tags), for the manually aligned reference corpus*

System	Configuration	mWER	BLEU
PB	Baseline	34.44	55.23
PB	+ RABCA	33.75	56.32
NB	Baseline	34.46	55.24
NB	+ RABCA	33.68	56.26

Table 3: *Results in the EuroParl Es2En task for the phrase-based system and the Ngram-based system.*

If we compare the performance of the algorithm in both systems, the quality of translation is improved in both cases. It is in the phrase-based system where the RABCA seems to slightly achieve a higher gain. Analysing the errors, we see that in some cases the Ngram-based baseline system has already performed a solution for the reorderings which are solved here, see (Crego et al., 2005).

Discussion. The RABCA infers better local reorderings than the ones provided only by the translation units (both phrases or tuples). Both measures, mWER and BLEU, improve significantly.
This task is relatively monotonic. Nevertheless, adding local reordering to the SMT systems leads to a better translation performance.

5. Conclusions

This paper has introduced a local statistical reordering approach which improves both a phrase and Ngram-based SMT systems. When dealing with local reorderings, better reorderings can be infered when comparing with the ones provided only by the translation units (both phrases or tuples). Experiments are reported in the EuroParl task (Spanish to English), but the algorithm can be used in any pair of languages. Results have shown that the RABCA algorithm improves the translation quality.

6. References

T. Brants. 2000. Tnt - a statistical part-of-speech tagger. *Proceedings of the Sixth Applied Natural Language Processing.*

X. Carreras, I. Chao, L. Padró, and M. Padró. 2004. Freeling: An open-source suite of language analyzers. In *4th Int. Conf. on Language Resources and Evaluation, LREC'06*, Lisboa, Portugal.

J. M. Crego, J. Mariño, and A. Gispert. 2005. Reordered search and tuple unfolding for ngram-based smt. *MT Summit X.*

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

A. de Gispert. 2005. Phrase linguistic classification for improving statistical machine translation. In *ACL 2005 Students Workshop*, Ann Arbor, USA, June.

S. Kanthak, D. Vilar, E. Matusov, R. Zens, and H. Ney. 2005. Novel reordering approaches in phrase-based statistical machine translation. In *Proceedings of the ACL Workshop on Building and Using Parallel Texts: Data-Driven Machine Translation and Beyond*, pages 167–174, Ann Arbor, MI, June.

P. Koehn, F.J. Och, and D. Marcu. 2003. Statistical phrase-based translation. In *Proc. of the Human Language Technology Conference, HLT-NAACL'2003*, pages 48–54, Edmonton, Canada, May.

P. Lambert, A. de Gispert, R. Banchs, and J. Mariño. 2006. Guidelines for word alignment and manual alignment. *Language Resources and Evaluation*, 39(4):267–285.

J.B. Mariño, R.E. Banchs, J.M. Crego, A. de Gispert, P. Lambert, J.A.R. Fonollosa, and M.R. Costa-jussà. 2006. N-gram based machine translation. *Computational Linguistics*, 32(4):527–549, December.

G.A. Miller, R. Beckwith, C. Fellbaum, D. Gross, K. Miller, and R. Tengi. 1991. Five papers on WordNet. *Special Issue of International Journal of Lexicography*, 3(4):235–312.

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

313.

S. Nießen and H. Ney. 2004. Statistical machine translation with scarce resources using morpho-syntactic information. *Computational Linguistics*, 30(2):181–204, June.

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

M. Popovic and H. Ney. 2006. Pos-based word reorderings for statistical machine translation. In *5th International Conference on Language Resources and Evaluation (LREC)*, pages 1278–1283, Genoa, May.

A. Stolcke. 2002. Srilm - an extensible language modeling toolkit. In *Proc. of the 7th Int. Conf. on Spoken Language Processing, ICSLP'02*, pages 901–904, Denver, USA, September.

C. Tillmann and T. Zhang. 2005. A localized prediction model for statistical machine translation. In *ACL*.

R. Zens, F.J. Och, and H. Ney. 2004. Improvements in phrase-based statistical machine translation. In *Proc. of the Human Language Technology Conference, HLT-NAACL'2004*, pages 257–264, Boston, MA (USA).