UPC-TALP Technical Report 2000-01
April 2009

A new subtree-transfer approach to syntax-based reordering for statistical machine translation

by

Maxim Khalilov and José A.R. Fonollosa
Universitat Politècnica de Catalunya
Campus Nord UPC, 08034
Barcelona, Spain

Mark Dras
Macquarie University
North Ryde NSW 2109
Sydney, Australia

TALP Reseach Center
Universitat Politècnica de Catalunya
2009

ABSTRACT

In this paper we address the problem of translating between languages with word order disparity. The idea of augmenting statistical machine translation (SMT) by using a syntax-based reordering step prior to translation, proposed in recent years, has been quite successful in improving translation quality. We present a new technique for extracting syntax-based reordering rules, which are derived through a syntactically augmented alignment of source and target texts.The parallel corpus with reordered source side is then passed to the N-gram-based machine translation system and the obtained results are contrasted with a monotone system performance.

In experiments, we show significant improvement for smaller Chinese-to-English BTEC translation task.

1 Introduction

One of the most challenging problems facing machine translation (MT) is how to place the translated words in order inherent in the target language. A monotone SMT system suffers from weakness in the distortion model, even if it is able to generate correct word-by-word translation. In this study we propose a reordering model that involves both source- and target-side syntax information in the word reordering process.

While a monotone translation approach is not able to deal with long-distance reorderings, a constituent tree structure contains this information which can be used, for example, to change the language topology scheme or clause restructuring.

Our work is inspired by the approach proposed in [IOS05], where the complete syntax-driven SMT system based on a two-side subtree transfer is described. In their approach they construct a probabilistic non-isomorphic tree mapping model based on a context-free breakdown of the source and target parse trees; extract alignment templates that incorporate the constraints of the parse trees; and apply syntax-based decoding. We propose to use a similar non-isomorphic subtree mapping to extract reordering rules, but instead of involving them directly in the translation process, we use them to monotonize the source portion of the bilingual corpus.

In the next step, the rules are applied to the source part of the same training corpus changing the source sentence structure such that it more closely matches the word order of the target language. Hence, the translation task is reformulated from the *plain source-to-target* to the *reordered source-to-target* translation, which makes a mutual word order closer to monotonic. It leads to a simplification of the translation task due to a shorter average length of bilingual units which it is more likely to see when translating an unseen set.

Local and long-range word reorderings are driven by automatically extracted permutation patterns operating with source language constituents and underlaid by non-isomorphic subtree transfer. The target-side parse tree utilization is optional: it is considered as a filter constraining the reordering rules to the set of patterns covered both by the source- and target-side subtrees. Apart from the reordering rules representing the order of child nodes, a set of additional rewrite rules based on a deep top-down subtree analysis is considered, which is another novel aspect of the paper.

We used the N-gram-based SMT system of [MBC$^+$06] to test the proposed syntax-based reordering model, which is an alternative to the phrase-based state-of-the-art Moses[1] system.

The rest of the paper is organized as follows: in Section 2 we outline the most significant related works, in Sections 3 the N-gram-based SMT system is briefly reviewed. Section 4 introduces the SBR technique, along with rules extraction and selection procedures. In Section 5 we present the results and contrast them with other reordering techniques, while Section 6 concludes the paper.

[1] www.statmt.org/moses/

2 Related work

There have been abundant publications on approaches involving context or additional information to solve the problem of word order disparity. In practice, a reordering model operates on a sentence level and is carried out based on word reordering rules derived from the training corpus. Reordering patterns can be purely statistical (see [CjF06], for example), use language-based syntactic information [CKK05], the reordering can be driven by a lattice of syntactically motivated alternative translations [Elm08] or be based on automatically extracted patterns driven by syntactical structure of the languages (see [CMn07b] as example). Another recent implementation of the preprocessing approach to syntax-based reordering though an n-best list generation can be found in [LZZ+07].

Word class-based reordering patterns were part of the Alignment Template system [OGK+04]. The modern state-of-the-art phrase-based translation system Moses, along with a distance based distortion model [KOM03], implements the reordering [TZ05], which is based on a so-called MSD (Monotone-Swap-Discontinuous) model, extracting reordering rules from a phrase alignment table.

Reordering algorithms specifically developed for an N-gram system include a constrained distance-based distortion model [CjMCdG+06], a linguistically motivated reordering model employing monotonic search graph extension [CMn07a] and a reordering model based on source-side dependency trees involvement in the refinement of monotonic reordering patterns [CMn07b].

An example of a word order monotonization strategy can be found in [CjF06], where a monotone sequence of source words is translated into the reordered sequence using SMT techniques. In theory, this approach intends to tackle a long-range reordering, however, in practice, a number of long-distance dependencies are not considered due to high sparseness of data.

In [XM04] the authors present a hybrid system for French-English translation, based on the principle of automatic rewrite patterns extraction using a parse tree and phrase alignments. This method differs from the one presented in this paper, among other distinctions, by a lexical model underlying the subtree syntax transfer (the one in this paper being the novel techniques inspired by [IOS05]) and a different statistical model used for translation (the authors conducted experiments on a phrase-based system, while we are concentrated on the experiments with the N-gram-based SMT).

Another important issue is the syntactic information incorporation into a purely SMT system. In [D.C05] an hireracically organized phrase-based model proposed, providing generalization of statistically extracted phrases with target-side syntactic categories [ZV06]. A representative sample of syntax-based systems include theoritecial MT systems based on bilingual synchronous grammar [Mel04] and parse tree-to-string translation models [YK01]. A comprehensive comparison of a phrase-based SMT system, following the Alignment Template approach [OGK+04] and a syntax-based string-to-tree model [GHKM04] can be found in [DKWM07].

3 Baseline SMT system

N-gram-based SMT has proved to be competitive with the state-of-the-art systems in recent evaluation campaigns [KHCj+08, LCjC+07].

According to the N-gram-based approach, the translation process is considered as an *arg max* searching for the translation hypothesis $\hat{e}_1^I$ maximizing a log-linear combination of a translation model (TM) and a set of feature models:

$$\hat{e}_1^I = \arg\max_{e_1^I} \left\{ \sum_{m=1}^{M} \lambda_m h_m(e_1^I, f_1^J) \right\} \tag{1}$$

where the feature functions h_m refer to the system models and the set of λ_m refers to the weights corresponding to these models.

The main difference between phrase-based and N-gram-based approaches lies in distinct representation of bilingual units, which are the components of the translation model (TM). While regular phrase-based SMT considers context only for phrase reordering but not for translation, the N-gram-based approach conditions translation decisions on previous translation decisions and operate with bilingual *n*-grams, so-called *tuples*, that are extracted from a word-to-word alignment.

Formally, it is expressed in form of three rules driving tuples extraction procedure:

- given a certain word-to-word alignment, a monotonic segmentation of each bilingual sentence pair is produced
- no word in a tuple is aligned to words outside of it
- no smaller tuples can be extracted without violating the previous constraints

Phrase-based SMT does not consider the last rule that, consequently, lead to a different representation of the bilingual context and need for constraints of the maximum phrase length.

Figure 1 shows example of tuple extraction resulting in four tuples. A detailed description of the N-gram-based approach can be found in [MBC+06].

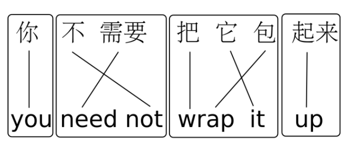


Figure 1: *Example of tuples extraction.*

3.1 Translation model

Bilingual **translation model** which is considered as an engine of the SMT system approximates the joint probability between source and target languages capturing bilingual context in form of standard n-grams, as follows:

$$p(S,T) = \prod_{k=1}^{K} p(\tilde{d}_k | \tilde{d}_{k-N+1}, ..., \tilde{d}_{k-1}) \tag{2}$$

where $\tilde{d} = (\tilde{s}, \tilde{t})$, s refers to source, t to target and $\tilde{d}_k$ to the k^{th} tuple of a given bilingual sentence pair segmented in K tuples.

3.2 Feature models

Along with a TM, the N-gram-based system implements a log-linear combination of 5 additional feature functions: *a target LM of words, a target LM of Part-of-Speech tags (POS), a word bonus model, a source-to-target* and *target-to-source lexicon models*.

3.3 Decoding and optimization

As decoder, we used MARIE[2] [CMndG05], a beam-search decoder implementing a distance-based constrained distortion model, limited by two parameters: *m* - a maximum distance measured number in words that a phrase can be reordered and *j* - a maximum number of "jumps" within a sentence [CjMCdG+06].

Given the development set and references, the log-linear combination of weights was adjusted using a simplex optimization method and an n-best re-ranking[3].

4 Syntax-based reordering

Our syntax-based reordering (SBR) system requires access to source and target language parse trees, along with the source-to-target and target-to-source word alignments intersection. In the framework of the study we used the Stanford Parser [KM03] for both languages, however the system permits using any other natural language parser allowing for different formal grammars for the source and the target languages.

4.1 Notation

SBR operates with source and target parse trees that represent the syntactic structure of a string in source and target languages according to a Context-Free Grammar (CFG).

[2] http://gps-tsc.upc.es/veu/soft/soft/marie/
[3] as described in http://www.statmt.org/jhuws/

This representation is called *"CFG form"*, and is formally defined in the usual way as $G = \langle N, T, R, S\rangle$, where N is a set of nonterminal symbols (corresponding to source-side phrase and part-of-speech tags); T is a set of source-side terminals (the lexicon), R is a set of production rules of the form $\eta \rightarrow \gamma$, with $\eta \in N$ and γ, which is a sequence of terminal and nonterminal symbols; and $S \in N$ is the distinguished symbol.

The reordering rules then have the form

$$\eta_0@0 \ldots \eta_k@k \longrightarrow \eta_{d_0}@d_0 \ldots \eta_{d_k}@d_k|Lexicon|p_1 \tag{3}$$

where $\eta_i \in N$ for all $0 \leq i \leq k$; $(d_o \ldots d_k)$ is a permutation of $(0 \ldots k)$; *Lexicon* includes the source-side set of words for each η_i; and p_1 is a probability associated with the rule. Figure 2 gives two examples of the rule format.

4.2 Rule extraction

Concept. Inspired by the ideas presented in [IOS05], where monolingual correspondences of syntactic nodes are used during decoding, we extract a set of bilingual patterns allowing for reordering as described below:

(1) align the monotone bilingual corpus with GIZA++[4] [ON03] and find the intersection of direct and inverse word alignments, resulting in the construction of the projection matrix P (see below));

(2) parse the source and the target parts of the parallel corpus;

(3) extract reordering patterns from the parallel non-isomorphic CFG-trees based on the word alignment intersection.

Step 2 is straightforward; we explain aspects of Steps 1 and 3 in more detail below. Figure 2 shows an example of the generation of two lexicalized rules; we use this below in our explanations.

Projection matrix. Bilingual content can be represented in the form of words or sequences of words depending on the syntactic role of the corresponding grammatical element (constituent or POS).

Given two parse trees and a word alignment intersection, a projection matrix P is defined as an $M \times N$ matrix such that M is the number of words in the target phrase; N is the number of words in the source phrase; and a cell (i, j) has a value based on the alignment intersection — this value is zero if word i and word j do not align, and is a unique non-zero link number if they do.

If a word that is aligned in only one direction appears in the brnch that is considered as a candidate to be involved into a reordering pattern, it does not appear in the the alignment projection matrix. For the trees in Figure 2,

$$P = \begin{pmatrix} 0 & 0 & 2 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \end{pmatrix}$$

[4] http://code.google.com/p/giza-pp/

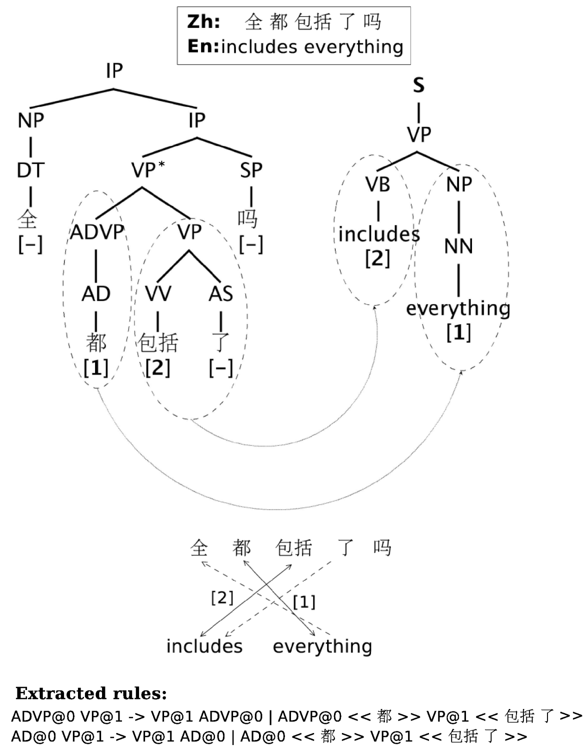


Figure 2: Example of reordering rules extraction.

Alignment and sub-trees interaction. Each non-terminal from the source and target parse trees is assigned a string carrying information about elements from the alignment intersection which are contained in its child nodes, taking into account the order of their appearance in the tree (AI). For example, the AI string assigned to the source-side internal node VP^* in Figure 2 is "1 2" and to the target-side VP is "2 1". This information is used to indicate the source-side nodes which are to be reordered according to the target language syntactical structure. Reordering patterns are extracted following the source and target-side AIs as shown in Figure 2 ("*main rules*").

If more than one non-zero element of the projection matrix is reachable through the child nodes, the AI has a more complex structure, providing information about elements from alignment intersection belonging to one or another child node. An example can be found in Figure 3.

Here, the subtree IP is assigned with the AI_{IP} = "1 (2 3)", meaning that it has two child node: the first contains the element 1 from the alignment intersection and the second - elements 2 and 3 (we call this subsequence "*closed*"). One-best reordering is kept at each node in the tree, and reach downwards as necessary. The reordering system considers nodes assigned with one or more children equally discerning the nodes with different order alignment elements.

Unary chains. Given an unary chains like "$ADVP \rightarrow AD \rightarrow ...$", rules are extracted for each level in this chain. For example in Figure 2, the directly extracted reordering rules are equivalent since the node $ADVP$ leads to the leaf through the node AD and does not have other edges.

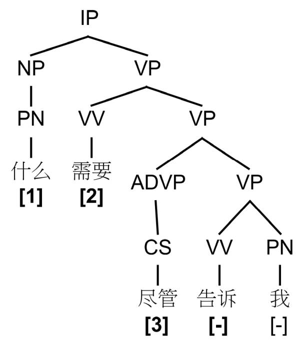


Figure 3: Example of complex AI structure.

The role of target-side parse tree. Roughly speaking, the use of target-side parse tree is optional. Although reordering is performed on the source side only, the target-side tree is of great importance: the reordering rules can be only extracted if the words covered by the rule are entirely covered by both a node in the source and in the target trees. It allows the more accurate determination of the covering and limits of the extracted rules.

4.3 Secondary rules

There are a lot of nodes for which a comparison of AIs indicates that a subtree transfer can be done, but segmentation of child nodes is not identical.

Figure 4 illustrates this situation. AI strings assigned to the root nodes of the trees contain the same elements, but segmentation and/or order of appearance of elements do not coincide. These subtrees can not be directly used for pattern extraction and more in-depth analysis is required.

We adopt the following six steps algorithm for each parent node from the source-side parse tree:

1. Find the AI sequence for the source-side top-level element (considering example, IP node is assigned as "(1 2) (3 4)").

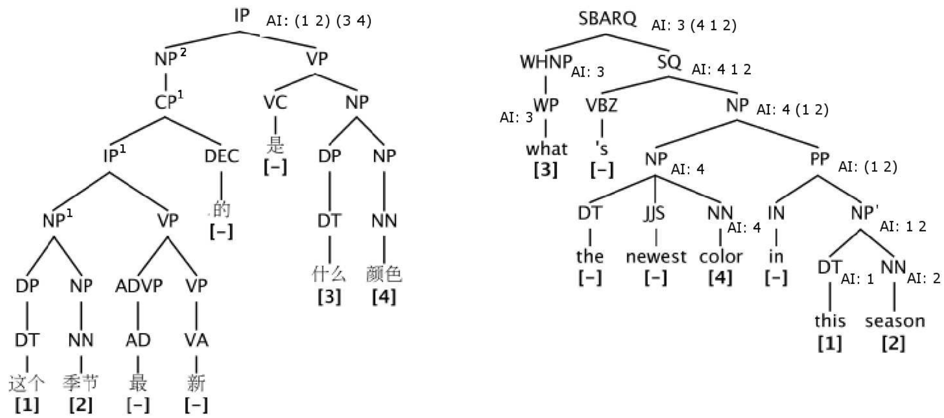
2. Go down through the target-side tree, finding AIs for each node.

3. Find all target-side closed subsequences for the source-side AI found on step 1. In example, it is the subsequence "(1 2)".

4. Find all target-side isolated nodes corresponding to the elements which were not covered on step 2. In example, these elements are "3" and "4".

5. Extend the set of source-side nodes found in steps 2 and 3 with equivalent branches. Since the words which are not presented in the alignment intersection does not affect the projection matrix, "equivalence" means here that all the branches spanning the elements from the given instance are considered equally (for example, elements NP^1 are equivalent to the nodes NP^1, IP^1, CP^1).

6. Place them in order corresponding to the target-side AI and construct the final reordering patterns (*"secondary rules"*).



Example of extracted rules:

```
NP@0 DP@1 NP@2 -> DP@1 NP@2 NP@0 | NP@ << 这个 季节 >> DP@1 << 什么 >> NP@2 << 颜色 >>
NP@0 DP@1 NN@2 -> DP@1 NN@2 NP@0 | NP@ << 这个 季节 >> DP@1 << 什么 >> NN@2 << 颜色 >>
NP@0 DT@1 NP@2 -> DT@1 NP@2 NP@0 | NP@ << 这个 季节 >> DT@1 << 什么 >> NP@2 << 颜色 >>
NP@0 DT@1 NN@2 -> DT@1 NN@2 NP@0 | NP@ << 这个 季节 >> DP@1 << 什么 >> NN@2 << 颜色 >>
CP@0 DP@1 NP@2 -> DP@1 NP@2 CP@0 | CP@0 << 这个 季节 最 新 的 >> DP@1 << 什么 >> NP@2 << 颜色 >>
CP@0 DP@1 NN@2 -> DP@1 NN@2 CP@0 | CP@0 << 这个 季节 最 新 的 >> DP@1 << 什么 >> NN@2 << 颜色>>
CP@0 DT@1 NP@2 -> DT@1 NP@2 CP@0 | CP@0 << 这个 季节 最 新 的 >> DT@1 << 什么 >> NP@2 << 颜色 >>
CP@0 DT@1 NN@2 -> DT@1 NN@2 CP@0 | CP@0 << 这个 季节 最 新 的 >> DT@1 << 什么 >> NN@2 << 颜色>>
NP@0 DP@1 NP@2 -> DP@1 NP@2 NP@0 | NP@0 << 这个 季节 最 新 的 >> DT@1 << 什么 >> NN@2 << 颜色>>
NP@0 DP@1 NN@2 -> DP@1 NN@2 NP@0 | NP@0 << 这个 季节 最 新 的 >> DT@1 << 什么 >> NN@2 << 颜色>>
NP@0 DT@1 NP@2 -> DT@1 NP@2 NP@0 | NP@0 << 这个 季节 最 新 的 >> DT@1 << 什么 >> NN@2 << 颜色>>
NP@0 DT@1 NN@2 -> DT@1 NN@2 NP@0 | NP@0 << 这个 季节 最 新 的 >> DT@1 << 什么 >> NN@2 << 颜色>>
...
NP@0 VP@1 -> VP@1 NP@0 | NP@0 << 这个 季节 最 新 的 >> VP@1 << 是 什么 颜色 >>
```

Figure 4: Example of “secondary“ rules extraction.

As illustration of the limitations incurred by target-side parse tree, the potential reordering pattern $NP@0\ VP@1 \rightarrow VP@1\ NP@0$ (referring to the top node in the Chinese tree) is not allowed due to distinct source- and target-side tree coverage.

4.4 Rule organization

Once the list of fully lexicalized reordering patterns is extracted, all the rules are progressively processed reducing amount of lexical information. Initial rules are iteratively expanded such that each element of the pattern is generalized until all the lexical elements of the rule are represented in the form of fully unlexicalized categories. Hence, from each initial pattern with N lexical elements, $2^N - 2$ partially lexicalized rules and 1 general rule are generated. An example of the process of delexicalization can be found in Figure 5.

Thus, finally three types of rules are available: (1) fully lexicalized (initial) rules, (2) partially lexicalized rules and (3) unlexicalized (general) rules.

```
Initial rule:
QP@0 CP@1 NP@2 -> QP@0 CP@1 NP@2 | QP0@ << 个 >>CP1@ << 不错 >>NP2@ << 夜总会 >>

Partially lexicalized rules:
QP@0 CP@1 NP@2 -> QP@0 CP@1 NP@2 | QP@0 << 个 >>CP@1 << NON >> NP@2 << NON >>
QP@0 CP@1 NP@2 -> QP@0 CP@1 NP@2 | QP@0 << NON >>CP@1 << 不错 >> NP@2 << NON >>
QP@0 CP@1 NP@2 -> QP@0 CP@1 NP@2 | QP@0 << NON >>CP@1 << NON >> NP@2 << 夜总会 >>
QP@0 CP@1 NP@2 -> QP@0 CP@1 NP@2 | QP@0 << NON >>CP@1 << 不错 >> NP@2 << 夜总会 >>
QP@0 CP@1 NP@2 -> QP@0 CP@1 NP@2 | QP@0 << 个 >>CP@1 << NON >> NP@2 << 夜总会 >>
QP@0 CP@1 NP@2 -> QP@0 CP@1 NP@2 | QP@0 << 个 >>CP1@ << 不错 >> NP@2 << NON >>

General rule:
QP@0 CP@1 NP@2 -> QP@0 CP@1 NP@2
```

Figure 5: Example of lexical rule expansion.

On the next step, the sets are processed separately: patterns are pruned and ambiguous rules are removed. Fully and partially lexicalized rules are not pruned out, but we set the thresholds k_{gener} to 3. All the rules from the corresponding set that appear less than k times are directly discarded. The probability of a pattern is estimated based on frequency in the training corpus, and only one the most probable rule is stored.

In the present version of the reordering system, only the one-best reordering is used in other stages of the algorithm, so the rule output functioning as an input to the next rule can lead to situations reverting the change of word order that the previously applied rule made. Therefore, the rules that can be ambiguous when applied sequentially during decoding are pruned according to the higher probability principle. For example, for the pair of patterns with the same lexicon (which is empty for a general rule leading to a recurring contradiction *NP@0 VP@1* → *VP@1 NP@0 p1*, *VP@0 NP@1* → *NP@1 VP@0 p2*), the less probable rule is removed.

Finally, there are three resulting parameter tables analogous to the "r-table" as stated in [YK01], consisting of POS- and constituent-based patterns allowing for reordering and monotone distortion.

4.5 Source-side monotonization

Rule application is performed as a bottom-up parse tree traversal following two principles:

(1) the longest possible rule is applied, i.e. among a set of nested rules, the rule with a longest left-side covering is selected. For example, in the case of the appearance of an *NN JJ RB* sequence and presence of the two reordering rules

NN@0 JJ@1 → ... and

NN@0 JJ@1 RB@2 → ...

the latter pattern will be applied.

(2) the rule containing the maximum lexical information is applied, i.e. in case there is more than one alternative pattern from different groups, the lexicalized rules have preference over the partially lexicalized, and partially lexicalized over general ones.

Figure 6 shows example of the reordered source-side tree corresponding to the example from Figure 2 with the applied pattern

$$ADVP@0\ VP@1 \rightarrow VP@1\ ADVP@0$$

and the given lexicon. The resulting reordered Chinese phrase more closely matches the order of the target language and is considered as a result of the subtree transfer.

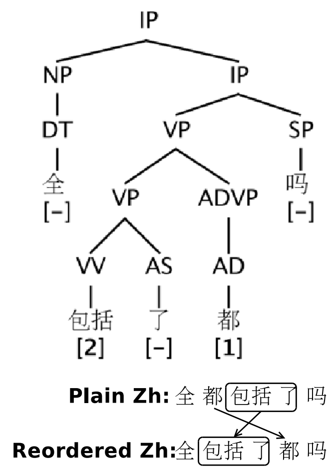


Figure 6: *Reordered source-side parse tree.*

Once the reordering of the training corpus is ready, it is realigned and new more monotonic alignment is passed to the SMT system. In theory, the word links from the original alignment can be used, however, due to our experience, running GIZA++ again results in a better word alignment since it is easier to learn on the modified training example.

5 Experiments and results

5.1 Corpus

The corpus that we used for training, development and testing is the Chinese-English BTEC speech corpus consisting of tourism-related sentences typically found in phrasebooks for tourists going abroad. The main reason why the Chinese-English translation task was chosen for experiments is that European languages are not so crucial for global (long-distance) reordering problem as the translation between Asian languages and English. Our motivation for BTEC corpus using is easiness and speed of experiment conduction, along with clearness of obtained results.

Basic statistics of the training material can be found in Table1.

The development and test datasets used to tune and test the system consist of 489 and 500 sentences, respectively, and are provided with 7 reference translations.

	Chinese	English
Sentences	44.9 K	44.9 K
Words	299.0 K	324.4 K
Vocabulary	11.4 K	9 K

Table 1: Basic statistics of the training corpus.

5.2 Experiment setup

Evaluation conditions were case-insensitive and with punctuation marks considered. We used the Stanford Parser as a NLP parsing engine [KM03] trained on the Chinese and English Penn Treebank sets (32 POS/44 constituent categories for Arabic Treebank and 48 POS/14 syntactic tags for English Treebank).

N-gram models were estimated using the SRILM toolkit [Sto02]. TM is represented in a 4-gram model form using modified Kneser-Ney discounting with interpolation, target language model (LM) of words is a 4-gram model with modified Kneser-Ney discounting, while a target-side POS LM is a 4-gram with Good-Turing backing-off.

For all system configurations, apart from monotone experiments, parameters of the distance-based reordering model were set to $m = 5$ and $j = 5$ for a trade-off between efficiency and accuracy.

The optimization criteria which was used in simplex optimization was the highest $4NIST + 100BLEU$ score (details about NIST metric are provided in [Dod02], BLEU score is described in [PRWZ02]).

5.3 Results

A number of unique rules (*rules*) for each of the three groups, along with a number of unique rules after processing and pruning as described in subsection 4.5 (*rules'*) can be found in Table 2.

	Main rules		Secondary rules	
	rules	rules'	rules	rules'
Lexicalized	31,176	3,688	1,589.157	1,479
Partially lexicalized	1,028,481	4,191	1,434,888	916
General	365	22	640,606	130

Table 2: Reordering rule statistics on the initial step and after pruning.

The following scores are reported in Table 3: final score obtained as a result of feature model weights tuning for development dataset (*dev*), BLEU and METEOR scores [BL05] for the test dataset. For the training set we present the number and vocabulary of tuples extracted from the monotone and reordered corpora.

We contrast four system configurations: (a.) no word reordering technique application on the preprocessing step, no distance-based distortion model (*Monotone*), (b.) SBR is applied involving main rules only, no distortion model applied (*SynBReor*), (c.) SBR is applied involving main rules only and allowing for distortion ($m = 5, j = 5$) during decoding (*SynBReor+mj*) and (d.) SBR is applied involving main and secondary rules and allowing for distortion ($m = 5, j = 5$) during decoding (*SynBReor+SecRules+mj*).

We also compare the obtained results with (e.) the constrained distortion model application to the monotone corpus (*Monotone+mj*), that allows comparing two techniques and demonstrates the effect of the algorithms application.

	dev	test BLEU	test METEOR	# tuples	voc tuples
Monotone	48.17	19.50	47.05	150,378	36,643
Monotone+mj	48.36	19.91	47.30	150,378	36,643
SynBReor	47.55	19.91	47.50	157,345	36,936
SynBReor+mj	49.35	20.69	47.83	157,345	36,936
SynBReor+SecRules+mj	47.83	19.70	47.52	141,430	36,501

Table 3: Summary of the experimental results.

5.4 Discussion

Application of the SBR technique demonstrates significant improvement in translation quality according to the automatic scores. The general trend is that evaluation metric on the test set improves with the reordering model complexity, although declining when the secondary rules are added.

SynBReor+mj is found to be the best system configuration, outperforming the monotone configuration by about 0.8 BLEU points (5.8 %) that is statistically significant for a 95% confidence interval and 1000 resamples [Koe04]. At first glance, the combination of these reordering techniques could introduce noise and hurt the results. However, the architecture of the distance-based model leads to a search space extension, with many more candidates; this helps in decoding, and does not interfere with the SBR, leading to a natural result improvement.

It is possible to see from Tables 2 and 3 that the introduction of secondary rules influences negatively the number of extracted tuples and comparing to the "main rules only" configuration shows a degradation in performance. Generally speaking, secondary rules include more elements than primary ones and are more difficult to be seen in the dataset parsed with the Stanford Parser. However, we speculate that accurate pruning of secondary rules could benefit the system performance significantly.

Finally, comparing a standard distance-based constrained distortion model coupled with decoding and SBR, the former shows better performance than the latter by 0.78 BLEU and 0.53 METEOR points that is still statistically significant for both metrics.

6 Conclusions and Future Work

In this paper we introduced a syntax-based reordering technique that monotonizes the word order of source and target languages involved in the process of bilingual unit extraction. As can be seen from the results presented, the proposed algorithm shows competitive performance comparing with an alternative fundamental distance-based reordering model.

The comparison is done on the smaller Chinese-English translation task with a strong need for word reorderings. Inspite of the fact that the major part of corpus sentences are short, there are some long sentences, demonstrating promising potential of the SBR algorithm (example can be found in Figure 7). On the next step we are planning to apply the presented reordering technique to a bigger Chinese-English corpus (NIST parallel corpus, for example).

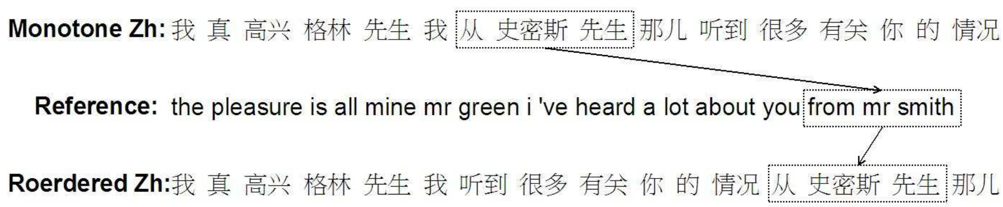


Figure 7: Example of SBR application.

The method achieves the same performance as a distance-based distortion model, and improves performance when combined with the latter. The use of syntax-based reorderings proves to be useful to improve translation accuracy for the task under consideration, however the incorporation of reordering rules, which are based on deep analysis of source and target parse trees (secondary rules) into the reordering system degrades system's performance. Nevertheless, we consider this feature to have potential given accurate tuning of pruning parameters, which will be future work.

The proposed approach is flexible and will be applied to the phrase-based systems. Apart from this task, further work includes the algorithm's application to a different language pair with distinct need for reorderings, analysis of the extracted tuples and development of the algorithm for accurate reordering rules selection.

References

[BL05] S. Banerjee and A. Lavie. METEOR: An automatic metric for MT evaluation with improved correlation with human judgments. In *Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*, pages 65–72, 2005.

[CjF06] M. R. Costa-jussà and J. A. R. Fonollosa. Statistical machine reordering. In *Proceedings of the HLT/EMNLP 2006)*, 2006.

[CjMCdG+06] M.R. Costa-jussà, J. M. Crego, A. de Gispert, P. Lambert, M. Khalilov, J. A. Fonollosa, J. B. Mariño, and R. E. Banchs. TALP phrase-based system and TALP system combination for IWSLT 2006. In *Proceedings of the IWSLT 2006*, pages 123–129, 2006.

[CKK05] M. Collins, Ph. Koehn, and I. Kučerová. Clause restructuring for statistical machine translation. In *Proceedings of the ACL'05*, pages 531–540, 2005.

[CMn07a] J. M. Crego and J. B. Mariño. Improving statistical MT by coupling reordering and decoding. *Machine Translation*, 20(3):199–215, 2007.

[CMn07b] J.M. Crego and J.B. Mariño. Syntax-enhanced N-gram-based SMT. In *Proceedings of the Machine Translation Summit (MT SUMMIT XI)*, 2007.

[CMndG05] J. M. Crego, J. B. Mariño, and A. de Gispert. An ngram-based statistical machine translation decoder. In *Proceedings of INTERSPEECH05*, 2005.

[D.C05] D.Chiang. A hierarchical phrase-based model for statistical machine translation. In *Proceedings of the Association for Computational Linguistics (ACL) 2005*, pages 263–270, 2005.

[DKWM07] S. DeNeefe, K. Knight, W. Wang, and D. Marcu. What can syntax-based MT learn from phrase-based MT? In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, pages 755–763, 2007.

[Dod02] G. Doddington. Automatic evaluation of machine translation quality using n-grams co-occurrence statistics. In *In HLT 2002 (Second Conference on Human Language Technology)*, pages 128–132, 2002.

[Elm08] J. Elming. Syntactic reordering integrated with phrase-based SMT. In *In Proc. of the ACL'08 Workshop SSST-2*, pages 46–54, Ohio, USA, June 2008.

[GHKM04] M. Galley, M. Hopkins, K. Knight, and D. Marcu. What's in a translation rule? In *Proceedings of the HLT/NAACL-04*, 2004.

[IOS05] K. Imamura, H. Okuma, and E. Sumita. Practical approach to syntax-based statistical machine translation. In *Proceedings of the 10th Machine Translation Summit (MT Summit X)*, pages 267–274, 2005.

[KHCj+08] M. Khalilov, A.H. Hernández, M.R. Costa-jussà, J.M. Crego, C.A. Henríquez, P. Lambert, J.A.R. Fonollosa, J.B. Mariño, and R. Banchs. The TALP-UPC ngram-based statistical machine translation system for ACL-WMT 2008. In *Proceedings of the ACL 2008 Third Workshop on Statistical Machine Translation (WMT'08)*, pages 127–131, 2008.

[KM03] D. Klein and C. Manning. Accurate unlexicalized parsing. In *Proceedings of the 41st Annual Meeting of the ACL 2003*, pages 423–430, 2003.

[Koe04] Ph. Koehn. Statistical significance tests for machine translation evaluation. In *Proceedings of Empirical Methods in Natural Language Processing (EMNLP) 2004*, pages 388–395, 2004.

[KOM03] Ph. Koehn, F. J. Och, and D. Marcu. Statistical phrase-based machine translation. In *Proceedings of the HLT-NAACL 2003*, pages 48–54, 2003.

[LCjC+07] P. Lambert, M.R. Costa-jussà, J. M. Crego, M. Khalilov, J. Mariño, R.E. Banchs, J.A.R. Fonollosa, and H. Shwenk. The TALP ngram-based SMT system for IWSLT 2007. In *Proceedings of the International Workshop on Spoken Language Translation (IWSLT07)*, pages 169–174, 2007.

[LZZ+07] C. Li, D. Zhang, M. Zhou, M. Li, and Y. Guan. A probabilistic approach to syntax-based reordering for statistical machine translation. In *Proceedings of the ACL'07*, pages 720–727, Prague, 2007.

[MBC+06] J. B. Mariño, R. E. Banchs, J. M. Crego, A. de Gispert, P. Lambert, J. A. R. Fonollosa, and M. R. Costa-jussà. N-gram based machine translation. *Computational Linguistics*, 32(4):527–549, December 2006.

[Mel04] I.D. Melamed. Statistical machine translation by parsing. In *Proceedings of the Association for Computational Linguistics (ACL) 2004*, pages 111–114, 2004.

[OGK+04] F. J. Och, D. Gildea, S. Khudanpur, A. Sarkar, K. Yamada, A. Fraser, S. Kumar, L. Shen, D. Smith, K. Eng, V. Jain, Z. Jin, and D. Radev. A Smorgasbord of Features for Statistical Machine Translation. In *Proceedings of HLT/NAACL04*, pages 161–168, 2004.

[ON03] F. Och and H. Ney. A systematic comparison of various statistical alignment models. *Computational Linguistics*, 29(1):19–51, 2003.

[PRWZ02] K. Papineni, S. Roukos, T. Ward, and W. Zhu. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics (ACL) 2002*, pages 311–318, 2002.

[Sto02] A. Stolcke. SRILM: an extensible language modeling toolkit. In *Proceedings of the Int. Conf. on Spoken Language Processing*, pages 901–904, 2002.

[TZ05] C. Tillmann and T. Zhang. A localized prediction model for statistical machine translation. In *Proceedings of the 43rd Annual Meeting on ACL 2005*, pages 557–564, 2005.

[XM04] F. Xia and M. McCord. Improving a statistical MT system with automatically learned rewrite patterns. In *Proceedings of the COLING 2004*, 2004.

[YK01] K. Yamada and K. Knight. A syntax-based statistical translation model. In *Proceedings of the Association for Computational Linguistics (ACL) 2001*, pages 523–530, 2001.

[ZV06] A. Zollmann and A. Venugopal. Syntax augmented machine translation via chart parsing. In *Proceedings of the North American Association for Computational Linguistics Conference (NAACL) 2006*, 2006.