Divide and Translate: Improving Long Distance Reordering in Statistical Machine Translation

Katsuhito Sudoh, Kevin Duh, Hajime Tsukada, Tsutomu Hirao, Masaaki Nagata
NTT Communication Science Laboratories
2-4 Hikaridai, Seika-cho, Soraku-gun, Kyoto 619-0237, Japan
sudoh@cslab.kecl.ntt.co.jp

Abstract

This paper proposes a novel method for long distance, clause-level reordering in statistical machine translation (SMT). The proposed method separately translates clauses in the source sentence and reconstructs the target sentence using the clause translations with non-terminals. The non-terminals are placeholders of embedded clauses, by which we reduce complicated clause-level reordering into simple word-level reordering. Its translation model is trained using a bilingual corpus with clause-level alignment, which can be automatically annotated by our alignment algorithm with a syntactic parser in the source language. We achieved significant improvements of 1.4% in BLEU and 1.3% in TER by using Moses, and 2.2% in BLEU and 3.5% in TER by using our hierarchical phrase-based SMT, for the English-to-Japanese translation of research paper abstracts in the medical domain.

1 Introduction

One of the common problems of statistical machine translation (SMT) is to overcome the differences in word order between the source and target languages. This *reordering* problem is especially serious for language pairs with very different word orders, such as English-Japanese. Many previous studies on SMT have addressed the problem by incorporating probabilistic models into SMT reordering. This approach faces the very large computational cost of searching over many possibilities, especially for long sentences. In practice the search can be made tractable by limiting its reordering distance, but this also renders long distance movements impossible. Some recent studies avoid the problem by reordering source words prior to decoding. This approach faces difficulties when the input phrases are long and require significant word reordering, mainly because their reordering model is not very accurate.

In this paper, we propose a novel method for translating long sentences that is different from the above approaches. Problematic long sentences often include embedded clauses[1] such as relative clauses. Such an embedded (subordinate) clause can usually be translated almost independently of words outside the clause. From this viewpoint, we propose a *divide-and-conquer* approach: we aim to translate the clauses separately and reconstruct the target sentence using the clause translations. We first segment a source sentence into clauses using a syntactic parser. The clauses can include non-terminals as placeholders for nested clauses. Then we translate the clauses with a standard SMT method, in which the non-terminals are reordered as words. Finally we reconstruct the target sentence by replacing the non-terminals with their corresponding clause translations. With this method, clause-level reordering is reduced to word-level reordering and can be dealt with efficiently. The models for clause translation are trained using a bilingual corpus with clause-level alignment. We also present an automatic clause alignment algorithm that can be applied to sentence-aligned bilingual corpora.

In our experiment on the English-to-Japanese translation of multi-clause sentences, the proposed method improved the translation performance by 1.4% in BLEU and 1.3% in TER by using Moses, and by 2.2% in BLEU and 3.5% in TER by using our hierarchical phrase-based SMT.

The main contribution of this paper is two-fold:

[1] Although various definitions of a *clause* can be considered, this paper follows the definition of "S" (sentence) in Enju. It basically follows the Penn Treebank II scheme but also includes SINV, SQ, SBAR. See http://www-tsujii.is.s.u-tokyo.ac.jp/enju/enju-manual/enju-output-spec.html#correspondence for details.

Proceedings of the Joint 5th Workshop on Statistical Machine Translation and MetricsMATR, pages 418–427,
Uppsala, Sweden, 11-16 July 2010.

1. We introduce the idea of explicit separation of in-clause and outside-clause reordering and reduction of outside-clause reordering into common word-level reordering.

2. We propose an automatic clause alignment algorithm, by which our approach can be used without manual clause-level alignment.

This paper is organized as follows. The next section reviews related studies on reordering. Section 3 describes the proposed method in detail. Section 4 presents and discusses our experimental results. Finally, we conclude this paper with our thoughts on future studies.

2 Related Work

Reordering in SMT can be roughly classified into two approaches, namely a search in SMT decoding and preprocessing.

The former approach is a straightforward way that models reordering in noisy channel translation, and has been studied from the early period of SMT research. Distance-based reordering is a typical approach used in many previous studies related to word-based SMT (Brown et al., 1993) and phrase-based SMT (Koehn et al., 2003). Along with the advances in phrase-based SMT, lexicalized reordering with a block orientation model was proposed (Tillmann, 2004; Koehn et al., 2005). This kind of reordering is suitable and commonly used in phrase-based SMT. On the other hand, a syntax-based SMT naturally includes reordering in its translation model. A lot of research work undertaken in this decade has used syntactic parsing for linguistically-motivated translation. (Yamada and Knight, 2001; Graehl and Knight, 2004; Galley et al., 2004; Liu et al., 2006). Wu (1997) and Chiang (2007) focus on formal structures that can be extracted from parallel corpora, instead of a syntactic parser trained using treebanks. These syntactic approaches can theoretically model reordering over an arbitrary length, however, long distance reordering still faces the difficulty of searching over an extremely large search space.

The preprocessing approach employs deterministic reordering so that the following translation process requires only short distance reordering (or even a monotone). Several previous studies have proposed syntax-driven reordering based on source-side parse trees. Xia and McCord (2004) extracted reordering rules automatically from bilingual corpora for English-to-French translation; Collins et al. (2005) used linguistically-motivated clause restructuring rules for German-to-English translation; Li et al. (2007) modeled reordering on parse tree nodes by using a maximum entropy model with surface and syntactic features for Chinese-to-English translation; Katz-Brown and Collins (2008) applied a very simple reverse ordering to Japanese-to-English translation, which reversed the word order in Japanese segments separated by a few simple cues; Xu et al. (2009) utilized a dependency parser with several hand-labeled precedence rules for reordering English to subject-object-verb order like Korean and Japanese. Tromble and Eisner (2009) proposed another reordering approach based on a linear ordering problem over source words without a linguistically syntactic structure. These preprocessing methods reorder source words close to the target-side order by employing language-dependent rules or statistical reordering models based on automatic word alignment. Although the use of language-dependent rules is a natural and promising way of bridging gaps between languages with large syntactic differences, the rules are usually unsuitable for other language groups. On the other hand, statistical methods can be applied to any language pairs. However, it is very difficult to reorder all source words so that they are monotonic with the target words. This is because automatic word alignment is not usually reliable owing to data sparseness and the weak modeling of many-to-many word alignments. Since such a reordering is not complete or may even harm word ordering consistency in the source language, these previous methods further applied reordering in their decoding. Li et al. (2007) used N-best reordering hypotheses to overcome the reordering ambiguity.

Our approach is different from those of previous studies that aim to perform both short and long distance reordering at the same time. The proposed method distinguishes the reordering of embedded clauses from others and efficiently accomplishes it by using a divide-and-conquer framework. The remaining (relatively short distance) reordering can be realized in decoding and preprocessing by the methods described above. The proposed framework itself does not depend on a certain language pair. It is based on the assumption that a source

language clause is translated to the corresponding target language clause as a continuous segment. The only language-dependent resource we need is a syntactic parser of the source language. Note that clause translation in the proposed method is a standard MT problem and therefore any reordering method can be employed for further improvement.

This work is inspired by syntax-based methods with respect to the use of non-terminals. Our method can be seen as a variant of tree-to-string translation that focuses only on the clause structure in parse trees and independently translates the clauses. Although previous syntax-based methods can theoretically model this kind of derivation, it is practically difficult to decode long multi-clause sentences as described above.

Our approach is also related to sentence simplification and is intended to obtain simple and short source sentences for better translation. Kim and Ehara (1994) proposed a rule-based method for splitting long Japanese sentences for Japanese-to-English translation; Furuse et al. (1998) used a syntactic structure to split ill-formed inputs in speech translation. Their splitting approach splits a sentence sequentially to obtain short segments, and does not undertake their reordering.

Another related field is clause identification (Tjong et al., 2001). The proposed method is not limited to a specific clause identification method and any method can be employed, if their clause definition matches the proposed method where clauses are independently translated.

3 Proposed Method

The proposed method consists of the following steps illustrated in Figure 1.

During training:

1) clause segmentation of source sentences with a syntactic parser (section 3.1)

2) alignment of target words with source clauses to develop a clause-level aligned corpus (section 3.2)

3) training the clause translation models using the corpus (section 3.3)

During testing:

1) clause translation with the clause translation models (section 3.4)

2) sentence reconstruction based on non-terminals (section 3.5)

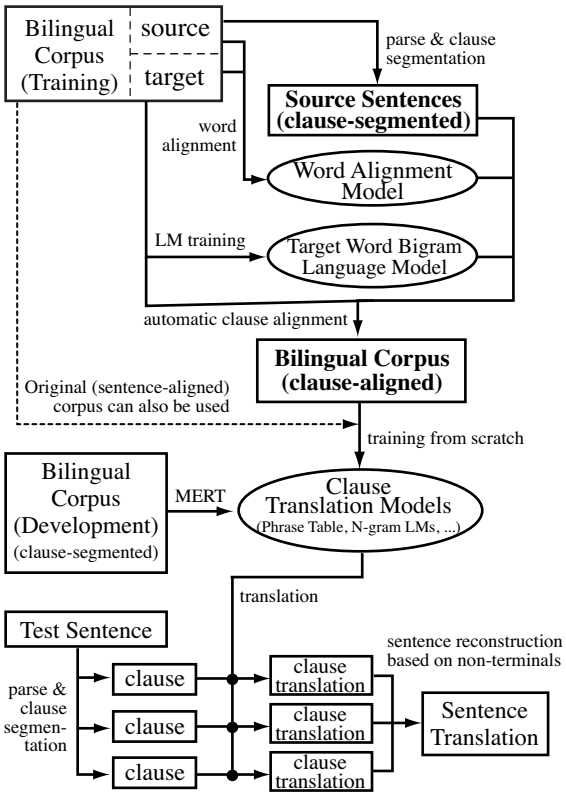

Figure 1: Overview of proposed method.

3.1 Clause Segmentation of Source Sentences

Clauses in source sentences are identified by a syntactic parser. Figure 2 shows a parse tree for the example sentence below. The example sentence has a relative clause modifying the noun *book*. Figure 3 shows the word alignment of this example.

English: *John lost the book that was borrowed last week from Mary.*

Japanese: *john wa* (topic marker) *senshu* (last week) *mary kara* (from) *kari* (borrow) *ta* (past tense marker) *hon* (book) *o* (direct object marker) *nakushi* (lose) *ta* (past tense marker) .

We segment the source sentence at the clause level and the example is rewritten with two clauses as follows.

- John lost the book *__s0* .
- that was borrowed last week from Mary

__s0 is a non-terminal symbol the serves as a placeholder of the relative clause. We allow an arbitrary

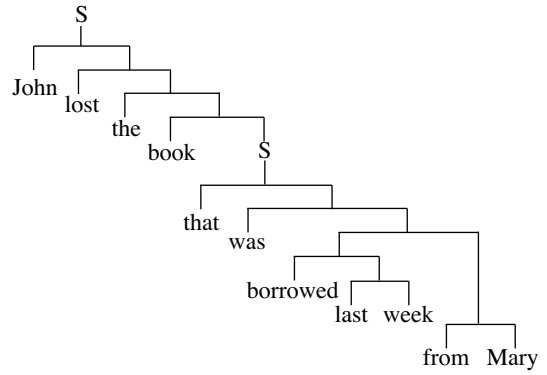

Figure 2: Parse tree for example English sentence. Node labels are omitted except S.

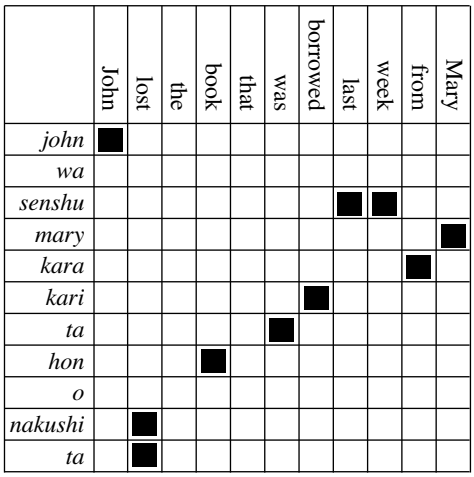

Figure 3: Word alignment for example bilingual sentence.

number of non-terminals in each clause[2]. A nested clause structure can be represented in the same manner using such non-terminals recursively.

3.2 Alignment of Target Words with Source Clauses

To translate source clauses with non-terminal symbols, we need models trained using a clause-level aligned bilingual corpus. A clause-level aligned corpus is defined as a set of parallel, bilingual clause pairs including non-terminals that represent embedded clauses.

We assume that a sentence-aligned bilingual corpus is available and consider the alignment of target words with source clauses. We can manually align these Japanese words with the English clauses as follows.

- *john wa __s0 hon o nakushi ta .*
  John lost the book *__s0* .
- *senshu mary kara kari ta*
  that was borrowed last week from Mary

Since the cost of manual clause alignment is high especially for a large-scale corpus, a natural question to ask is whether this resource can be obtained from a sentence-aligned bilingual corpus *automatically with no human input*. To answer this, we now describe a simple method for dealing with clause alignment data from scratch, using only the word alignment and language model probabilities inferred from bilingual and monolingual corpora.

Our method is based on the idea that automatic clause alignment can be viewed as a classification problem: for an English sentence with N words ($\mathbf{e} = (e_1, e_2, \ldots, e_N)$) and K clauses ($\tilde{\boldsymbol{e}}^1$,$\tilde{\boldsymbol{e}}^2$,...,$\tilde{\boldsymbol{e}}^K$), and its Japanese translation with M words ($\mathbf{f} = (f_1, f_2, \ldots, f_M)$), the goal is to classify each Japanese word into one of $\{1, \ldots, K\}$ classes. Intuitively, the probability that a Japanese word f_m is assigned to class $k \in \{1, \ldots, K\}$ depends on two factors:

1. The probability of translating f_m into the English words of clause k (i.e. $\sum_{e \in \tilde{\mathbf{e}}^k} p(e|f_m)$). We expect f_m to be assigned to a clause where this value is high.

2. The language model probability (i.e. $p(f_m|f_{m-1})$). If this value is high, we expect f_m and f_{m-1} to be assigned to the same clause.

We implement this intuition using a graph-based method. For each English-Japanese sentence pair, we construct a graph with K *clause nodes* (representing English clauses) and M *word nodes* (representing Japanese words). The edge weights between word and clause nodes are defined as the sum of lexical translation probabilities $\sum_{e \in \tilde{\mathbf{e}}^k} p(e|f_m)$. The edge weights between words are defined as the bigram probability $p(f_m|f_{m-1})$. Each clause node is labeled with a class ID $k \in \{1, \ldots, K\}$. We then *propagate* these K labels along the graph to label the M word nodes. Figure 4 shows the graph for the example sentence.

Many label propagation algorithms are available. The important thing is to use an algorithm that encourages node pairs with strong edge weights to receive the same label. We use the label propagation algorithm of (Zhu et al., 2003). If we

[2] In practice not so many clauses are embedded in a single sentence but we found some examples with nine embedded clauses for coordination in our corpora.

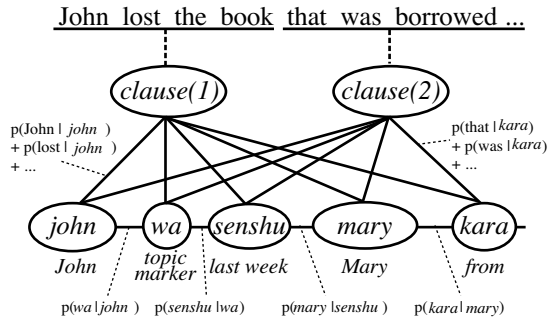

Figure 4: Graph-based representation of the example sentence. We propagate the clause labels to the Japanese word nodes on this graph to form the clause alignments.

assume the labels are binary, the following objective is minimized:

$$\underset{\boldsymbol{l}\in\mathcal{R}^{K+M}}{\operatorname{argmin}} \sum_{i,j} w_{ij}(l_i - l_j)^2 \qquad (1)$$

where w_{ij} is the edge weight between nodes i and j ($1 \leq i \leq K+M$, $1 \leq j \leq K+M$), and $\boldsymbol{l}$ ($l_i \in \{0,1\}$) is a vector of labels on the nodes. The first K elements of $\boldsymbol{l}$, $\boldsymbol{l}_c = (l_1, l_2, ..., l_K)^T$, are constant because the clause nodes are pre-labeled. The remaining M elements, $\boldsymbol{l}_f = (l_{K+1}, l_{K+2}, ..., l_{K+M})^T$, are unknown and to be determined. Here, we consider the decomposition of the weight matrix $\boldsymbol{W} = [w_{ij}]$ into four blocks after the K-th row and column as follows:

$$\boldsymbol{W} = \begin{bmatrix} \boldsymbol{W}_{cc} & \boldsymbol{W}_{cf} \\ \boldsymbol{W}_{fc} & \boldsymbol{W}_{ff} \end{bmatrix} \qquad (2)$$

The solution of eqn. (1), namely $\boldsymbol{l}_f$, is given by the following equation:

$$\boldsymbol{l}_f = (\boldsymbol{D}_{ff} - \boldsymbol{W}_{ff})^{-1}\boldsymbol{W}_{fc}\,\boldsymbol{l}_c \qquad (3)$$

where $\boldsymbol{D}$ is the diagonal matrix with $d_i = \sum_j w_{ij}$ and is decomposed similarly to $\boldsymbol{W}$. Each element of $\boldsymbol{l}_f$ is in the interval $(0,1)$ and can be regarded as the label propagation probability. A detailed explanation of this solution can be found in Section 2 of (Zhu et al., 2003). For our multi-label problem with K labels, we slightly modified the algorithm by expanding the vector $\boldsymbol{l}$ to an $(M+K)\times K$ binary matrix $\boldsymbol{L} = [\,\boldsymbol{l}_1\,\boldsymbol{l}_2\,...\,\boldsymbol{l}_K\,]$.

After the optimization, we can normalize $\boldsymbol{L}_f$ to obtain the clause alignment scores $t(l_m = k|f_m)$ between each Japanese word f_m and English clause k. Theoretically, we can simply output the clause id k' for each f_m by finding $k' = \arg\max_k t(l_m = k|f_m)$. In practice, this may sometimes lead to Japanese clauses that have too many gaps, so we employ a two-stage procedure to extract clauses that are more contiguous.

First, we segment the Japanese sentence into K clauses based on a dynamic programming algorithm proposed by Malioutov and Barzilay (2006). We define an $M \times M$ similarity matrix $\boldsymbol{S} = [s_{ij}]$ with $s_{ij} = \exp(-||\boldsymbol{l}^i - \boldsymbol{l}^j||)$ where $\boldsymbol{l}^i$ is $(K+i)$-th row vector in the label matrix $\boldsymbol{L}$. s_{ij} represents the similarity between the i-th and j-th Japanese words with respect to their clause alignment score distributions; if the score distributions are similar then s_{ij} is large. The details of this algorithm can be found in (Malioutov and Barzilay, 2006). The clause segmentation gives us contiguous Japanese clauses $\tilde{\boldsymbol{f}}^1$, $\tilde{\boldsymbol{f}}^2$, ..., $\tilde{\boldsymbol{f}}^K$, thus minimizing inter-segment similarity and maximizing intra-segment similarity. Second, we determine the clause labels of the segmented clauses, based on clause alignment scores $\boldsymbol{T} = [T_{kk'}]$ for English and automatically-segmented Japanese clauses:

$$T_{kk'} = \sum_{f_m \in \tilde{\boldsymbol{f}}_{k'}} t(l_m = k|f_m) \qquad (4)$$

where $\tilde{\boldsymbol{f}}_{k'}$ is the j'-th Japanese clause. In descending order of the clause alignment score, we greedily determine the clause label [3].

3.3 Training Clause Translation Models

We train clause translation models using the clause-level aligned corpus. In addition we can also include the original sentence-aligned corpus. We emphasize that we can use standard techniques for heuristically extracted phrase tables, word n-gram language models, and so on.

3.4 Clause Translation

By using the source language parser, a multi-clause source sentence is reduced to a set of clauses. We translate these clauses with a common SMT method using the clause translation models.

Here we present another English example *I bought the magazine which Tom recommended yesterday.* This sentence is segmented into clauses as follows.

[3] Although a full search is available when the number of clauses is small, we employ a greedy search in this paper.

- I bought the magazine *__s0* .
- which Tom recommended yersterday

These clauses are translated into Japanese:

- *watashi* (I) *wa* (topic marker) *__s0*
  zasshi (magazine) *o* (direct object marker)
  kat (buy) *ta* (past tense marker).
- tom *ga* (subject marker) *kino* (yesterday)
  susume (recommend) *ta* (past tense marker)

3.5 Sentence Reconstruction

We reconstruct the target sentence from the clause translations, based on non-terminals. Starting from the clause translation of the top clause, we recursively replace non-terminal symbols with their corresponding clause translations. Here, if a non-terminal is eventually deleted in SMT decoding, we simply concatenate the translation behind its parent clause.

Using the example above, we replace the non-terminal symbol *__s0* with the second clause and obtain the Japanese sentence:
watashi wa tom ga kino susume ta zasshi o kat ta .

4 Experiment

We conducted the following experiments on the English-to-Japanese translation of research paper abstracts in the medical domain. Such technical documents are logically and formally written, and sentences are often so long and syntactically complex that their translation needs long distance reordering. We believe that the medical domain is suitable as regards evaluating the proposed method.

4.1 Resources

Our bilingual resources were taken from the medical domain. The parallel corpus consisted of research paper abstracts in English taken from PubMed[4] and the corresponding Japanese translations.

The training portion consisted of 25,500 sentences (*no-clause-seg.*; original sentences without clause segmentation). 4,132 English sentences in the corpus were composed of multiple clauses and were separated at the clause level by the procedure in section 3.1. As the syntactic parser, we used the Enju[5] (Miyao and Tsujii, 2008) English HPSG parser. For these training sentences, we automatically aligned Japanese words with each English clause as described in section 3.2 and developed a clause-level aligned corpus, called *auto-aligned* corpus. We prepared manually-aligned (oracle) clauses for reference, called *oracle-aligned* clauses. The clause alignment error rate of the auto-aligned corpus was 14% (number of wrong clause assignments divided by total number of words). The development and test portions each consisted of 1,032 multi-clause sentences. because this paper focuses only on multi-clause sentences. Their English-side was segmented into clauses in the same manner as the training sentences, and the development sentences had oracle clause alignment for MERT.

We also used the Life Science Dictionary[6] for training. We extracted 100,606 unique English entries from the dictionary including entries with multiple translation options, which we expanded to one-to-one entries, and finally we obtained 155,692 entries.

English-side tokenization was obtained using Enju, and we applied a simple preprocessing that removed articles (a, an, the) and normalized plural forms to singular ones. Japanese-side tokenization was obtained using MeCab[7] with ComeJisyo[8] (dictionary for Japanese medical document tokenization). Our resource statistics are summarized in Table 1.

4.2 Model and Decoder

We used two decoders in the experiments, Moses[9] (Koehn et al., 2007) and our in-house hierarchical phrase-based SMT (almost equivalent to Hiero (Chiang, 2007)). Moses used a phrase table with a maximum phrase length of 7, a lexicalized reordering model with `msd-bidirectional-fe`, and a distortion limit of 12[10]. Our hierarchical phrase-based SMT used a phrase table with a maximum rule length of 7 and a window size (Hiero's Λ) of 12 [11]. Both

[4] http://www.ncbi.nlm.nih.gov/pubmed/

[5] http://www-tsujii.is.s.u-tokyo.ac.jp/enju/index.html

[6] http://lsd.pharm.kyoto-u.ac.jp/en/index.html

[7] http://mecab.sourceforge.net/

[8] http://sourceforge.jp/projects/comedic/ (in Japanese)

[9] http://www.statmt.org/moses/

[10] Unlimited distortion was also tested but the results were worse.

[11] A larger window size could not be used due to its memory requirements.

Table 1: Data statistics on training, development, and test sets. All development and test sentences are multi-clause sentences.

Training			
Corpus Type	#words		#sentences
Parallel (no-clause-seg.)	E	690,536	25,550
	J	942,913	
Parallel (auto-aligned)	E	135,698	4,132 (10,766 clauses)
	J	183,043	
(oracle-aligned)	J	183,147	
Dictionary	E	263,175	155.692 (entries)
	J	291,455	
Development			
Corpus Type	#words		#sentences
Parallel (oracle-aligned)	E	34,417	1,032 (2,683 clauses)
	J	46,480	
Test			
Corpus Type	#words		#sentences
Parallel (clause-seg.)	E	34,433	1,032 (2,737 clauses)
	J	45,975	

decoders employed two language models: a word 5-gram language model from the Japanese sentences in the parallel corpus and a word 4-gram language model from the Japanese entries in the dictionary. The feature weights were optimized for BLEU (Papineni et al., 2002) by MERT, using the development sentences.

4.3 Compared Methods

We compared four different training and test conditions with respect to the use of clauses in training and testing. The development (i.e., MERT) conditions followed the test conditions. Two additional conditions with oracle clause alignment were also tested for reference.

Table 2 lists the compared methods. First, the proposed method (*proposed*) used the auto-aligned corpus in training and clause segmentation in testing. Second, the baseline method (*baseline*) did not use clause segmentation in either training or testing. Using this standard baseline method, we focused on the advantages of the divide-and-conquer translation itself. Third, we tested the same translation models as used with the proposed method for test sentences without clause segmentation, (*comp.(1)*). Although this comparison method cannot employ the proposed clause-level reordering, it was expected to be better than the baseline method because its translation model can be trained more precisely using the finely aligned clause-level corpus. Finally, the second comparison method (*comp.(2)*) translated segmented clauses with the baseline (without clause segmentation) model, as if each of them was a single sentence. Its translation of each clause was expected to be better than that of the baseline because of the efficient search over shortened inputs, while its reordering of clauses (non-terminals) was unreliable due to the lack of clause information in training. Its sentence reconstruction based on non-terminals was the same as with the proposed method. Although non-terminals in the second comparison method were out-of-vocabulary words and may be deleted in decoding, all of them survived and we could reconstruct sentences from translated clauses throughout the experiments. In addition, two other conditions were tested: using oracle-aligned clauses in training: the proposed method trained using oracle-aligned (*oracle*) clauses and the first comparison method using oracle-aligned (*oracle-comp.*) clauses.

4.4 Results

Table 3 shows the results in BLEU, Translation Edit Rate (TER) (Snover et al., 2006), and Position-independent Word-error Rate (PER) (Och et al., 2001), obtained with Moses and our hierarchical phrase-based SMT, respectively. Bold face results indicate the best scores obtained with the compared methods (excluding oracles).

The proposed method consistently outperformed the baseline. The BLEU improvements with the proposed method over the baseline and comparison methods were statistically significant according to the bootstrap sampling test ($p < 0.05$, 1,000 samples) (Zhang et al., 2004). With Moses, the improvement when using the proposed method was 1.4% (33.19% to 34.60%) in BLEU and 1.3% (57.83% to 56.50%) in TER, with a slight improvement in PER (35.84% to 35.61%). We observed: $oracle \gg proposed \gg comp.(1) \gg baseline \gg comp.(2)$ by the Bonferroni method, where the symbol $A \gg B$ means "A's improvement over B is statistically significant." With the hierarchical phrase-based SMT, the improvement was 2.2% (32.39% to 34.55%) in BLEU, 3.5% (58.36% to 54.87%) in TER, and 1.5% in PER (36.42% to 34.79%). We observed: $oracle \gg proposed \gg$

Table 2: Compared methods.

Test \ Training	w/ auto-aligned	w/o aligned	w/ oracle-aligned
clause-seg.	**proposed**	comp.(2)	oracle
no-clause-seg.	comp.(1)	baseline	oracle-comp.

$\{comp.(1), comp.(2)\} \gg baseline$ by the Bonferroni method. The oracle results were better than these obtained with the proposed method but the differences were not very large.

4.5 Discussion

We think the advantage of the proposed method arises from three possibilities: 1) better translation model training using the fine-aligned corpus, 2) an efficient decoder search over shortened inputs, and 3) an effective clause-level reordering model realized by using non-terminals.

First, the results of the first comparison method (comp.(1)) indicate an advantage of the translation models trained using the auto-aligned corpus. The training of the translation models, namely word alignment and phrase extraction, is difficult for long sentences due to their large ambiguity. This result suggests that the use of clause-level alignment provides fine-grained word alignments and precise translation models. We can also expect that the model of the proposed method will work better for the translation of single-clause sentences.

Second, the average and median lengths (including non-terminals) of the clause-seg. test set were 13.2 and 10 words, respectively. They were much smaller than those of no-clause-seg. at 33.4 and 30 words and are expected to help realize an efficient SMT search. Another observation is the relationship between the number of clauses and translation performance, as shown in Figure 5. The proposed method achieved a greater improvement in sentences with a greater number of clauses. This suggests that our divide-and-conquer approach works effectively for multi-clause sentences. Here, the results of the second comparison method (comp.(2)) with Moses were worse than the baseline results, while there was an improvement with our hierarchical phrase-based SMT. This probably arose from the difference between the decoders when translating out-of-vocabulary words. The non-terminals were handled as out-of-vocabulary words under the comp.(2) condition.

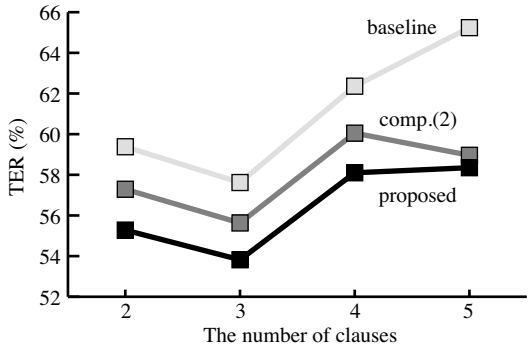

Figure 5: Relationship between TER and number of clauses for proposed, baseline, and comp.(2) when using our hierarchical phrase-based SMT.

Moses generated erroneous translations around such non-terminals that can be identified at a glance, while our hierarchical phrase-based SMT generated relatively good translations. This may be a decoder-dependent issue and is not an essential problem.

Third, the results obtained with the proposed method reveal an advantage in reordering in addition to the previous two advantages. The difference between the PERs with the proposed method and the baseline with Moses was small (0.2%) in spite of the large differences in BLEU and TER (about 1.5%). This suggests that the proposed method is better in word ordering and implies our method is also effective in reordering. With the hierarchical phrase-based SMT, the proposed method showed a large improvement from the baseline and comparison methods, especially in TER which was better than the best Moses configuration (proposed). This suggests that the decoding of long sentences with long-distance reordering is not easy even for the hierarchical phrase-based SMT due to its limited window size, while the hierarchical framework itself can naturally model a long-distance reordering. If we try to find a derivation with such long-distance reordering, we will probably be faced with an intractable search space and computation time. Therefore, we can conclude that the proposed divide-and-

Table 3: Experimental results obtained with Moses and our hierarchical phrase-based SMT, in BLEU, TER, and PER.

Moses : BLEU (%) / TER (%) / PER (%)			
Training / Test	w/ auto-aligned	w/o aligned	w/ oracle-aligned
clause-seg.	**34.60 / 56.50** / 35.61	32.14 / 58.78 / 36.08	35.31 / 55.12 / 34.42
no-clause-seg.	34.22 / 56.90 / **35.20**	33.19 / 57.83 / 35.84	34.24 / 56.67 / 35.03
Hierarchical : BLEU (%) / TER (%) / PER (%)			
Training / Test	w/ auto-aligned	w/o aligned	w/ oracle-aligned
clause-seg.	**34.55 / 54.87 / 34.79**	33.03 / 56.70 / 36.03	35.08 / 54.22 / 34.77
no-clause-seg.	33.41 / 57.02 / 35.86	32.39 / 58.36 / 36.42	33.83 / 56.26 / 34.96

conquer approach provides more practical long-distance reordering at the clause level.

We also analyzed the difference between automatic and manual clause alignment. Since auto-aligned corpus had many obvious alignment errors, we suspected these noisy clauses hurt the clause translation model. However, they were not serious in terms of final translation performance. So we can conclude that our proposed divide-and-conquer approach is promising for long sentence translation. Although we aimed to see whether we could bootstrap using existing bilingual corpora in this paper, we imagine better clause alignment can be obtained with some supervised classifiers.

One problem with the divide-and-conquer approach is that its independently-translated clauses potentially cause disfluencies in final sentence translations, mainly due to wrong inflections. A promising solution is to optimize a whole sentence translation by integrating search of each clause translation but this may require a much larger search space for decoding. More simply, we may be able to approximate it using n-best clause translations. This problem should be addressed for further improvement in future studies.

5 Conclusion

In this paper we proposed a clause-based divide-and-conquer approach for SMT that can reduce complicated clause-level reordering to simple word-level reordering. The proposed method separately translates clauses with non-terminals by using a well-known SMT method and reconstructs a sentence based on the non-terminals, to reorder long clauses. The clause translation models are trained using a bilingual corpus with clause-level alignment, which can be obtained with an unsupervised graph-based method using sentence-aligned corpora. The proposed method improves the translation of long, multi-clause sentences and is especially effective for language pairs with large word order differences, such as English-to-Japanese.

This paper focused only on clauses as segments for division. However, other long segments such as prepositional phrases are similarly difficult to reorder correctly. The divide-and-conquer approach itself can be applied to long phrases, and it is worth pursuing such an extension. As another future direction, we must develop a more sophisticated method for automatic clause alignment if we are to use the proposed method for various language pairs and domains.

Acknowledgments

We thank the U. S. National Library of Medicine for the use of PubMed abstracts and Prof. Shuji Kaneko of Kyoto University for the use of Life Science Dictionary. We also thank the anonymous reviewers for their valuable comments.

References

Peter F. Brown, Stephen A. Della Pietra, Vincent J. Della Pietra, and Robert L. Mercer. 1993. The mathematics of statistical machine translation: Parameter estimation. *Computational Linguistics*, 19(2):263–311.

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

Michael Collins, Philipp Koehn, and Ivona Kuĉerová. 2005. Clause restructuring for statistical machine translation. In *Proc. ACL*, pages 531–540.

Osamu Furuse, Setsuo Yamada, and Kazuhide Yamamoto. 1998. Splitting long or ill-formed input for robust spoken-language translation. In *Proc. COLING-ACL*, pages 421–427.

Michel Galley, Mark Hopkins, Kevin Knight, and Daniel Marcu. 2004. What's in a translation rule? In *Proc. NAACL*, pages 273–280.

Jonathan Graehl and Kevin Knight. 2004. Training tree transducers. In *Proc. HLT-NAACL*, pages 105–112.

Jason Katz-Brown and Michael Collins. 2008. Syntactic reordering in preprocessing for Japanese-English translation: MIT system description for NTCIR-7 patent translation task. In *Proc. NTCIR-7*, pages 409–414.

Yeun-Bae Kim and Terumasa Ehara. 1994. A method for partitioning of long Japanese sentences with subject resolution in J/E machine translation. In *Proc. International Conference on Computer Processing of Oriental Languages*, pages 467–473.

Phillip Koehn, Franz Josef Och, and Daniel Marcu. 2003. Statistical phrase-based translation. In *Proc. HLT-NAACL*, pages 263–270.

Philipp Koehn, Amittai Axelrod, Alexandra Birch Mayne, Chris Callison-Burch, Miles Osborne, and David Talbot. 2005. Edinburgh system description for the 2005 IWSLT speech translation evaluation. In *Proc. IWSLT*.

Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, Chris Dyer, Ondrej Bojar, Alexandra Constantin, and Evan Herbst. 2007. Moses: Open source toolkit for statistical machine translation. In *Proc. ACL Companion Volume Proceedings of the Demo and Poster Sessions*, pages 177–180.

Chi-Ho Li, Dongdong Zhang, Mu Li, Ming Zhou, Minghui Li, and Yi Guan. 2007. A probabilistic approach to syntax-based reordering for statistical machine translation. In *Proc. ACL*, pages 720–727.

Yang Liu, Qun Liu, and Shouxun Lin. 2006. Tree-to-String alignment template for statistical machine translation. In *Proc. Coling-ACL*, pages 609–616.

Igor Malioutov and Regina Barzilay. 2006. Minimum cut model for spoken lecture segmentation. In *Proc. Coling-ACL*, pages 25–32.

Yusuke Miyao and Jun'ichi Tsujii. 2008. Feature forest models for probabilistic HPSG parsing. *Computational Linguistics*, 34(1):35–80.

Franz Josef Och, Nicola Ueffing, and Hermann Ney. 2001. An efficient A* search algorithm for statistical machine translation. In *Proc. the ACL Workshop on Data-Driven Methods in Machine Translation*, pages 55–62.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei Jing Zhu. 2002. BLEU: a method for automatic evaluation of machine translation. In *Proc. ACL*, pages 311–318.

Matthew Snover, Bonnie Dorr, Richard Schwartz, Linnea Micciulla, and John Makhoul. 2006. A study of translation edit rate with targeted human annotation. In *Proc. AMTA*, pages 223–231.

Christoph Tillmann. 2004. A unigram orientation model for statistical machine translation. In *Proc. HLT-NAACL*, pages 101–104.

Erik F. Tjong, Kim Sang, and Hervé Déjean. 2001. Introduction to the CoNLL-2001 shared task: Clause identification. In *Proc. CoNLL*, pages 53–57.

Roy Tromble and Jason Eisner. 2009. Learning linear ordering problems for better translation. In *Proc. EMNLP*, pages 1007–1016.

Dekai Wu. 1997. Stochastic inversion transduction grammars and bilingual parsing of parallel corpora. *Computational Linguistics*, 23(3):377–404.

Fei Xia and Michael McCord. 2004. Improving a statistical MT system with automatically learned rewrite patterns. In *Proc. COLING*, pages 508–514.

Peng Xu, Jaeho Kang, Michael Ringgaard, and Franz Och. 2009. Using a dependency parser to improve SMT for Subject-Object-Verb languages. In *Proc. HLT-NAACL*, pages 245–253.

Kenji Yamada and Kevin Knight. 2001. A syntax-based statistical translation model. In *Proc. ACL*, pages 523–530.

Ying Zhang, Stephan Vogel, and Alex Weibel. 2004. Interpreting BLEU/NIST scores: How much improvement do we need to have a better system? In *Proc. LREC*, pages 2051–2054.

Xiaojin Zhu, Zoubin Ghahramani, and John Lafferty. 2003. Semi-supervised learning using gaussian fields and harmonic functions. In *Proc. ICML*, pages 912–919.