Source Sentence Reordering for Phrase-based Machine Translation Systems

Author(s) Name(s)
Author Affiliation(s)
E-mail

Abstract

We introduce a method for learning to reorder source sentences. In our approach, sentences are transformed into new sequences of words aimed at reducing non-local reorderings in phrase translation. The method involves automatically extracting instances of structural divergences from sentence pairs, and automatically learning lexicalized grammatical rules probabilistically encoded with bilingual word-order relations. At run-time, source sentences are reordered by applying the rules prior to phrase-based machine translation systems. Experiments show that our method cleanly captures systematic similarities and differences in languages' grammars, resulting in substantial improvement over state-of-the-art phrase-based translation systems.

1. Introduction

A myriad of Machine Translation (MT) models (also known as decoders) have been proposed to translate source (e.g., English) sentences into target (e.g., Chinese) sentences. Among them is phrase-based model, fundamental yet influential to the field of MT.

Phrase-based systems typically treat sentences as sequences of phrases and translate them without the knowledge of word orders in languages, leading to incorrect few or no changes in phrase orders at the target side. Consider the English sentence "They are still being detained by the police". A good Chinese translation from Google Translate is "他們 仍 被 警方 扣留" with the Chinese counterpart of the word "detained", i.e. "扣留", and that of the phrase "by the police", i.e. "被 警方", reordered. However, provided with another similar sentence "They are still being questioned by the police", Google Translate *does not* correctly order the translated phrases and returns a confusing translation "他們 仍然 受到 質疑 警察". Aforementioned two translations differ in accuracy and readability, which could be explained by the lack of generality in phrase translation pairs: phrase-based systems do not exploit structural aspects of languages such as word-order regularities or grammars, and put too much emphasis on exact wording. Intuitively, by pre-ordering the syntactic structures of source sentences to match those of target language, the reordering demand on subsequent phrase-based systems could be reduced and better translations might be acquired.

We present a model that automatically learns to transform source-language (SL) sentences into ones expected to share similar word orientations with target-language (TL) sentences via lexicalized syntactic reordering rules. The translation of a sentence transformed by our reordering model is shown in Figure 1. Our model learns reordering rules automatically during training by analyzing word-aligned source-parsed sentence pairs.

At run-time, our model determines the orientations (e.g., straightness or inversion) of the tree nodes in SL parse trees and reorders them accordingly. The reordered sentences can be used as input to word aligners (e.g., GIZA++) or as input to MT systems, in view of alleviating potentially negative impact of structural divergences on system performance.

2. Related Work

Many researches have focused on modeling word distortion or structural reordering during translation. Some condition distortion distribution on absolute word positions ([3]), on relative positions of phrases ([6,9]), and even on source words ([1]). Some determine structural orders using synchronous rules ([4] and Bracketing Transduction Grammar (BTG) rules in [12]). And some exploit monolingual (i.e., SL or TL) parse trees to capture languages' word-order preferences ([17,5,8]).

Recently, [18], [11] and [16] presents a version of BTG where head words of the structural constituents, function words, source-side syntax structures are used as reordering evidence, respectively. [19], an extension

(a) Before Reordering:
[$_S$[$_{NP_1}$They/PRP][$_{VP_1}$are/VBP[$_{ADVP}$still/RB] [$_{VP_2}$being/VBG[$_{VP_3}$questioned/VBN[$_{PP}$by/IN[$_{NP_2}$the/DT police/NN]]]]]]
(b) After Reordering:
[$_S$[$_{NP_1}$They/PRP][$_{VP_1}$are/VBP[$_{ADVP}$still/RB] [$_{VP_2}$being/VBG[$_{VP_3}$[$_{PP}$by/IN[$_{NP_2}$the/DT police/NN]]questioned/VBN]]]]
(c) Chinese Translation:
他們 現 正 由 警方 訊問

Figure 1. An English sentence before and after our syntactic reordering model. The Chinese translation is obtained by submitting sentence in (b) to Google Translate. The subscripts of linguistic symbols are for easy future reference.

to [4], further incorporates linguistic knowledge to weight the synchronous rules.

In a study related to our work, [14] uses rewrite patterns based on dependency-motivated parse trees to reorder source sentences. The number of the extracted rewrite patterns is exponential to that of words. In contrast, ours is linear to the number of words. Recently, [13] describes a model which transforms SL sentences to fit TL word orders via reordering rules. The main difference from our current work is that their rules are hand crafted without lexical information and reorderings are strictly required when rules are encountered. In this paper, we introduce a method for *automatically* learning lexicalized syntactic reordering rules. The significance of lexical and grammatical items in our rules, and associated orientation (i.e., straightness or inversion) probabilities are estimated by discriminative learning approach such as maximum entropy or conditional random fields.

3. The Method

We focus on preprocessing SL sentences prior to translation. Our goal is to produce reordered source sentences expected to have little reordering demand on subsequent phrase-based systems such that more grammatical translations could be acquired. We now formally state the problem that we are addressing.

Problem Statement: We are given a general purpose phrase-based translation system *TS*, a SL syntactic parser *SP*, and a source sentence *e*. Our goal is to obtain the translation of *e* via *TS* that is likely to match TL word-order regularities. For this, we apply syntactic reordering rules to transform *e* into *e'* that is likely to lead to more fluent and grammatical translation.

3.1. Learning Reordering Rules

In the first stage of the learning process, we first parse the SL sentences in bilingual corpus *C* via syntactic parser *SP*, then align the bitexts using word aligner *WA*, and at last filter out sentence pairs whose word-aligning ratios are not high enough to be though of as reliable candidates for training. $C_{\text{remained}}=\{(e, f, \pi_e)\}$ where π_e stands for the parse tree of the source sentence *e* and *f* for the target sentence, and its word alignment result *WAR* are delivered to the next stage.

procedure GenerateTrainingCases(C_{remained},*WAR*)
(1) *InstCol*=“”
for each (e, f, π_e) in C_{remained}
(2) *aInst*= “” // a training instance
(3) *TLSpan*=FindTLSpan(f, π_e,*WAR*)
(4) for each contiguous word w_l and w_r in e
(5a) *Ancestor*=FindClsCommonAncestor(w_l, w_r, π_e)
(5b) D_{lhs}=FindImmDescendent(*Ancestor*, w_l, π_e)
(5c) D_{rhs}=FindImmDescendent(*Ancestor*, w_r, π_e)
if *TLSpan*[D_{lhs}].max < *TLSpan*[D_{rhs}].min
(6) append ((*Ancestor*, D_{lhs}, w_l, D_{rhs}, w_r), “S”) to *aInst*
if *TLSpan*[D_{lhs}].min > *TLSpan*[D_{rhs}].max
(7) append ((*Ancestor*, D_{lhs}, w_l, D_{rhs}, w_r), “I”) to *aInst*
add *aInst* to *InstCol*
return *InstCol*

Figure 2. Extracting word-order relations.

TLSpan[PRP]=(1,1)	*TLSpan*[NP_1]=(1,1)
TLSpan[VBP]=(2,2)	*TLSpan*[ADVP]=(3,3)
TLSpan[RB]=(3,3)	*TLSpan*[NP_2]=(5,5)
TLSpan[VBG]=(NULL,NULL)	*TLSpan*[PP]=(4,5)
TLSpan[VBN]=(6,6)	*TLSpan*[VP_3]=(4,6)
TLSpan[IN]=(4,4)	*TLSpan*[VP_2]=(4,6)
TLSpan[DT]=(NULL,NULL)	*TLSpan*[VP_1]=(2,6)
TLSpan[NN]=(5,5)	*TLSpan*[S]=(1,6)

Figure 3. Sample output on TL spans of source nodes. Nodes of source words are omitted.

In the second stage, we leverage the algorithm in Figure 2 to collect training instances for word orders in involved languages. In Step (1) of the algorithm, we define an instance collection to gather TL structural orders with respect to SL ones. For each sentence pair, we construct a training instance (Step (2)) and identify the TL span of each SL tree node based on SL parse tree and word alignments (Step (3)). The TL span of a node is denoted as a binary tuple with the first element indicating the leftmost TL word position this node can cover (i.e., minimum index) and the second indicating the rightmost (i.e., maximum index). Sample output of Step (3), using the parsed English sentence in Figure 1(a) and its translation "他們 現 仍 被 警方 訊問", is shown in Figure 3. TL spans provide information on how the target language orders the corresponding source constituents.

In Step (4) consecutive words in sentence *e* represented by w_l and w_r are recognized as boundary words of two adjacent phrasal constituents. Following [15], we also use boundary words as reordering evidence. In Step (5a) we traverse the parse tree to identify the closest common ancestor *Ancestor*, covering the abovementioned two adjacent phrases, of words w_l and w_r while in Step (5b)/(5c) we denote the immediate left-hand-side/right-hand-side descendant of *Ancestor* along the path down to w_l/w_r as D_{lhs}/D_{rhs}. Syntactic constituents D_{lhs} and D_{rhs} border on words w_l and w_r. Afterwards, we examine how target language orders counterparts of D_{lhs} and D_{rhs} by heuristically comparing *TLSpan*[D_{lhs}] and *TLSpan*[D_{rhs}]. In Step (6) and (7) we annotate the attribute tuple (*Ancestor*,D_{lhs},w_l,D_{rhs},w_r), depicting our lexicalized syntactic rules, as "S" (i.e., straightness) and "I" (i.e., inversion) respectively. Note that the boundary words are also included in the attribute tuple. The rationale is better explained by the examples in Figure 4.

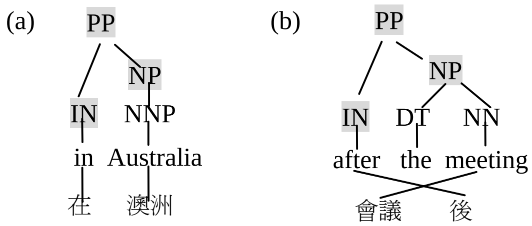

Figure 4. Reordering examples.

While the Chinese translations in Figure 4(a) are in order with the English words, similar sub-tree (constituents are shown in shade) has crossing alignment links in Figure 4(b), implying English and Chinese reversely construct the corresponding phrases. Since syntactic sub-trees with coarse-grained linguistic symbols (e.g., Penn tags) are likely to be the same, our model further exploits lexical contents to impose desirable word orders.

In the final stage of training, we employ conditional random fields model (CRFs) to estimate the significances of lexical and grammatical items in rules in determining TL word orders, and to estimate orientation (i.e., straightness or inversion) probabilities.

3.2. Run-Time Source Sentence Reordering

At run-time, for each source parse tree π_e, we first collect its attribute tuples (i.e., (*Ancestor*,D_{lhs},w_l,D_{rhs},w_r) in Figure 2) and then use the trained CRFs to find the most probable sequence of orientation labels (i.e., "S" or "I") of the tuples. Notice that the labels may be inter-dependent.

Based on the labeling result, the reordered source sentence expected to have TL-like word orders can be obtained. The sentence is then fed to phrase-based decoders for translation (see Figure 1).

4. Experiments

4.1. Data Sets and Experimental Settings

We used the first 200K sentence pairs of the news portion of Hong Kong Parallel Text (LDC2004T08) as our bilingual corpus *C*. We syntactically parsed the English (i.e., source) end via Berkeley parser. GIZA++ was applied on *C* to obtain word alignment. In the process of learning context-sensitive reordering rules, alignment ratio was set to 0.8. An implementation of CRFs, CRF++ (http://crfpp.sourceforge.net/), was used in training and at run-time.

4.2. Evaluation Results

Our English-to-Chinese MT training data (whole 200K sentence pairs) and MT testing data (1035 English sentences of average 28 words chosen from news portion of Hong Kong Parallel Text, excluding training data) were pre-arranged by our model as: $\mathbf{U_{tr}}$, the original training set; $\mathbf{R_{tr}}$, the reordered training set; $\mathbf{U_{ts}}$, the original test set; $\mathbf{R_{ts}}$, the reordered test set. To compare among these training and test sets, the same evaluation flow was adopted: GIZA++ imposing word alignments on $\mathbf{U_{tr}}$ or $\mathbf{R_{tr}}$, followed by construction of phrase table and translating $\mathbf{U_{ts}}$ or $\mathbf{R_{ts}}$ using Pharaoh ([7]), and finally evaluating translation quality measured by BLEU ([10]).

Applying $\mathbf{U_{tr}}$ and $\mathbf{U_{ts}}$ to the above flow constitutes our baseline. Utilizing $\mathbf{R_{tr}}$ and $\mathbf{U_{ts}}$ amounts to an inspection of the effect of syntactic reordering on word alignment quality (yet evaluated in the context of MT) while utilizing $\mathbf{U_{tr}}$ and $\mathbf{R_{ts}}$ an inspection on how non-local structural divergences might influence the performance of the phrase-based system, Pharaoh.

Table 1. Results on translation quality.

	$\mathbf{U_{tr}}$	$\mathbf{R_{tr}}$
$\mathbf{U_{ts}}$	(1) 23.43	(2) 24.16
$\mathbf{R_{ts}}$	(3) 24.76	(4) **25.71**

As suggested by the translation results in Table 1, when exploiting the reordered training sentences produced by our model to perform word alignment, the translation quality increased by 0.7 BLEU point ((2) vs. (1)). If we simply reordered the test sentences, we yielded a substantial improvement of 1.3 BLEU points over translating the original test sentences ((3) vs. (1)). Encouragingly, when source sentence reordering was applied to both sets of data, we got the most benefit out of the proposed approach: almost 2.3 BLEU-point increases ((4) vs. (1)).

For comparison, Table 2 shows the translation quality of three systems: Moses[1] using data sets of $\mathbf{U_{tr}}$ and $\mathbf{U_{ts}}$; (Wang et al., 2007)[+] which utilizes manual *unlexicalized* syntactic reordering rules in [13] *and* manual *lexicalized* ones to acquire its own $\mathbf{R_{tr}}$ and $\mathbf{R_{ts}}$ (the underlying decoder is Pharaoh); our reordering model. Our model achieved an absolute gain of 1.7 BLEU points over Moses with a complicated reordering model. Moreover, the performance of our system was comparable to that of (Wang et al., 2007)[+], indicating that the simple automatic learning procedure in Section 3.1 led to similar bilingual word-order analyses with human exports'.

Table 2. System comparison on MT task.

System	Data used	BLEU
Moses	$\mathbf{U_{tr}}$+$\mathbf{U_{ts}}$	23.94
(Wang et al., 2007)[+]	$\mathbf{R_{tr}}$+$\mathbf{R_{ts}}$	25.37
Ours	$\mathbf{R_{tr}}$+$\mathbf{R_{ts}}$	**25.71**

5. Future Work and Summary

Many avenues exist for future research and improvement of our system. Potential features (e.g., heights, source spans and head words of tree nodes and semantic classes of source words) could be incorporated into our model. Additionally, an interesting direction to explore is the applicability of our model in other distantly-related language pairs such as English and Japanese as well as English and Arabic.

In summary, we have introduced a model for automatically capturing structural divergences of involved languages using discriminative model of CRFs. In the evaluation, we have shown that phrase-based decoder benefits from our source sentence reordering model which cleanly learns similar word-order regularities with human exports' analyses.

[1] Its reordering model was configured to msd-bidirectional-fe, a sophisticated lexicalized reordering model.

6. References

1. Yaser Al-Onaizan and Kishore Papineni, "Distortion Model for Statistical Machine Translation", *ACL*, 2006, pp. 529-536.
2. Necip F. Ayan and Bonnie J. Dorr, "Going Beyond AER: An Extensive Analysis of Word Alignments and Their Impact on MT", *ACL*, 2006, pp. 9-16.
3. Peter F. Brown, Stephen A. Della Pietra, Vincent J. Della Pietra, and Robert L. Mercer, "The Mathematics of Statistical Machine Translation: Parameter Estimation", *Computational Linguistics*, 1993, 19(2), pp. 263-311.
4. David Chiang, "A Hierarchical Phrase-based Model for Statistical Machine Translation", *ACL*, 2005, pp. 263-270.
5. Michel Galley, Jonathan Graehl, Kevin Knight, Daniel Marcu, Steve DeNeefe, Wei Wang, and Ignacio Thayer, "Scalable Inference and Training of Context-Rich Syntactic Translation Models", *ACL*, 2006, pp. 961-968.
6. Philipp Koehn, Franz Och, and Daniel Marcu, "Statistical Phrase-based Translation", *NAACL*, 2003, pp. 48-54.
7. Philipp Koehn, "Pharaoh: A Beam Search Decoder for Phrase-based Statistical Machine Translation", *AMTA*, 2004, pp. 115-124.
8. Yang Liu, Yun Huang, Qun Liu, and Shouxun Lin, "Forest-to-string Statistical Translation Rules", *ACL*, 2007, pp. 704-711.
9. Franz J. Och and Hermann Ney, "The Alignment Template Approach to Statistical Machine Translation", *Computational Linguistics*, 2004, 30(4), pp. 417-449.
10. Kishore Papineni, Salim Roukos, Todd Ward and Wei-Jing Zhu, "BLEU: A Method for Automatic Evaluation of Machine Translation", *ACL*, 2002, pp. 311-318.
11. Hendra Setiawan, Min-Yen Kan and Haizhou Li, "Ordering Phrases with Function Words", 2007, *ACL*, pp. 712-719.
12. Dekai Wu, "Stochastic Inversion Transduction Grammars and Bilingual Parsing of Parallel Corpora", *Computational Linguistics*, 1997, 23(3), pp. 377-403.
13. Chao Wang, Michael Collins and Philipp Koehn, "Chinese Syntactic Reordering for Statistical Machine Translation", *EMNLP*, 2007, pp. 737-745.
14. Fei Xia and Michael McCord, "Improving A Statistical MT System with Automatically Learned Rewrite Patterns", *COLing*, 2004, pp. 508-514.
15. Deyi Xiong, Qun Liu, and Shouxun Lin, "Maximum Entropy Based Phrase Reordering Model for Statistical Machine Translation", *ACL*, 2006, pp. 521-528.
16. Deyi Xiong, Min Zhang, Aiti Aw, and Haizhou Li, "Linguistically Annotated BTG for Statistical Machine Translation", *COLing*, 2008, pp. 1009-1016.
17. Kenji Yamada and Kevin Knight, "A Decoder for Syntax-based Statistical MT", *ACL*, 2002, pp. 303-310.
18. H. Zhang and D. Gildea, "Stochastic Lexicalized Inversion Transduction Grammar for Alignment", *ACL*, 2005, pp. 475-482.
19. Dongdong Zhang, Mu Li, Chi-Ho Li and Ming Zhou, "Phrase Reordering Model Integrating Syntactic Knowledge for SMT", *EMNLP*, 2007, pp. 533-540