Generalizing Local and Non-Local Word-Reordering Patterns for Syntax-Based Machine Translation

Bing Zhao
IBM T.J. Watson Research
Yorktown Heights, NY-10598
zhaob@us.ibm.com

Yaser Al-onaizan
IBM T.J. Watson Research
Yorktown Heights, NY-10598
onaizan@us.ibm.com

Abstract

Syntactic word reordering is essential for translations across different grammar structures between syntactically distant language-pairs. In this paper, we propose to embed local and non-local word reordering decisions in a synchronous context free grammar, and leverages the grammar in a chart-based decoder. Local word-reordering is effectively encoded in Hiero-like rules; whereas non-local word-reordering, which allows for long-range movements of syntactic chunks, is represented in tree-based reordering rules, which contain variables correspond to source-side syntactic constituents. We demonstrate how these rules are learned from parallel corpora. Our proposed shallow Tree-to-String rules show significant improvements in translation quality across different test sets.

1 Introduction

One of the main issues that a translator (human or machine) must address during the translation process is how to match the different word orders between the source language and the target language. Different language-pairs require different levels of word reordering. For example, when we translate between English and Spanish (or other Romance languages), most of the word reordering needed is local because of the shared syntactical features (e.g., Spanish noun modifier constructs are written in English as modifier noun). However, for syntactically distant language-pairs such as Chinese-English, long-range reordering is required where whole phrases are moved across the sentence.

The idea of "*syntactic cohesion*" (Fox, 2002) is characterized by its simplicity, which has attracted researchers for years. Previous works include several approaches of incorporating syntactic information to *preprocess* the source sentences to make them more like the target language in structure. Xia and McCord (2004) (Niessen and Ney, 2004; Collins et al., 2005) described approaches applied to language-pairs such as French-English and German-English. Later, Wang et al. (2007) presented specific rules to pre-order long-range movements of words, and improved the translations for Chinese-to-English. Overall, these works are similar, in that they design a few language-specific and linguistically motivated reordering rules, which are generally simple. The *eleven* rules described in Wang et al. (2007) are appealing, as they have rather simple structure, modeling only NP, VP and LCP via one-level sub-tree structure with two children, in the source parse-tree (a special case of ITG (Wu, 1997)). It effectively enhances the quality of the phrase-based translation of Chinese-to-English. One major weakness is that the reordering decisions were done in the preprocessing step, therefore rendering the decoding process unable to recover the reordering errors from the rules if incorrectly applied to. Also the reordering decisions are made without the benefits of additional models (e.g., the language models) that are typically used during decoding.

Another method to address the re-ordering problem in translation is the Hiero model proposed by Chiang (2005), in which a probabilistic synchronous context free grammar (PSCFG) was applied to guide the decoding. Hiero rules generalize phrase-pairs

Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing, pages 572–581,
Honolulu, October 2008.

by introducing a single generic nonterminal (i.e., a variable) [X]. The combination of variables and lexicalized words in a Hiero rule nicely captures local word and phrase reordering (modeling an implicit reordering window of max-phrase length). These rules are then applied in a CYK-style decoder. In Hiero rules, any nested phrase-pair can be generalized as variables [X]. This usually leads to too many redundant translations, which worsens the *spurious ambiguities* (Chiang, 2005) problems for both decoding and optimization (i.e., parameter tuning). We found that*variables* (*nonterminal* [X]) in Hiero rules offer a generalization too coarse to improve the effectiveness of hierarchical models' performance.

We propose to enrich the variables in Hiero rules with additional source syntactic reordering information, in the form of shallow Tree-to-String syntactic structures. The syntactic information is represented by flat one-level sub-tree structures, with Hiero-like nonterminal variables at the leaf nodes. The syntactic rules, proposed in this paper, are composed of (possibly lexicalized) source treelets and target surface strings, with one or more variables that help capture local-reordering similar to the Hiero rules. Variables in a given rule are derived not only from the embedded aligned blocks (phrase-pairs), but also from the aligned source syntactic constituents. The aligned constituents, as in our empirical observations for Chinese-English, tend to move together in translations. The decoder is guided by these rules to reduce spurious derivations; the rules also constrain the exploration of the search space toward better translation quality and sometime improved speed by breaking long sentences into pieces. Overall, what we want is to enable the long-range reordering decisions to be local in a chart-based decoder.

To be more specific, we think the simple shallow syntactic structure is powerful enough for capturing the major structure-reordering patterns, such as NP, VP and LCP structures. We also use simple frequency-based feature functions, similar to the blocks used in phrase-based decoder, to further improve the rules' representation power. Overall, this enables us to avoid either a complex decoding process to generate the source parse tree, or difficult combinatorial optimizations for the feature functions associated with rules.

In Marton and Resnik (2008), hiero variables were disambiguated with additional binary feature functions, with their weights optimized in standard MER training. The combinatorial effects of the added feature functions can make the feature selection and optimization of the weights rather difficult. Since the grammar is essentially the same as the Hiero ones, a standard CYK decoder can be simply applied in their work. Word reordering can also be addressed via distortion models. Work in (Al-Onaizan and Kishore, 2006; Xiong et al., 2006; Zens et al., 2004; Kumar and Byrne, 2005; Tillmann and Zhang, 2005) modeled the limited information available at phrase-boundaries. Syntax-based approaches such as (Yamada and Knight, 2001; Graehl and Knight, 2004; Liu et al., 2006) heavily rely on the parse-tree to constrain the search space by assuming a strong mapping of structures across distant language-pairs. Their algorithms are also subject to parsers' performances to a larger extent, and have high complexity and less scalability in reality. In Liu et al. (2007), multi-level tree-structured rules were designed, which made the decoding process very complex, and auxiliary rules have to be designed and incorporated to shrink multiple source nonterminals into one target nonterminal. From our empirical observations, most of the time, however, the multi-level tree-structure is broken in the translation process, and POS tags are frequently distorted. Indeed, strictly following the source parse tree is usually not necessary, and maybe too expensive for the translation process.

The remainder of this paper is structured as follows: in section § 2, we define the notations in our synchronous context free grammar, in section § 3, the rule extractions are illustrated in details, in section § 4, the decoding process of applying these rules is described. Experiments in § 5 were carried out using GALE Dev07 datasets. Improved translation qualities were obtained by applying the proposed Tree-to-String rules. Conclusions and discussions are given in § 6.

2 Shallow Tree-to-String Rules

Our proposed rules are in the form of probabilistic synchronous context free grammar (PSCFG). We adopt the notations used in (Chiang, 2005). Let N be a set of nonterminals, a rule has the following

form:

$$X \rightarrow < \ell; \gamma; \alpha; \sim; \bar{w} >, \quad (1)$$

where X abstracts nonterminal symbols in N; $\gamma \in [N, V_S]^+$ is a sequence of one or more source [1] words (as in the vocabulary of V_S) and nonterminal symbols in N; $\alpha \in [N, V_T]^+$ is a sequence of one or more target words (in V_T) and nonterminals in N . $\sim$ is the one-to-one alignment of the nonterminals between γ and α; $\bar{w}$ contains non-negative weights associated with each rule; ℓ is a label-symbol specifying the root node of the source span covering γ. In our grammar, ℓ is one of the labels (e.g., NP) defined in the source treebank tagset (in our case UPenn Chinese tagset) indicating that the source span γ is rooted at ℓ. Additionally, a NULL tag $\emptyset$ in ℓ denotes a flat structure of γ, in which no constituent structure was found to cover the span, and we need to back off to the normal Hiero-style rules. Our nonterminal symbols include the labels and the POS tags in the source parse trees.

In the following, we will illustrate the Tree-to-String rules we are proposing. At the same time, we will describe the extraction algorithm, with which we derive our rules from the word-aligned source-parsed parallel text. Our nonterminal set N is a reduced set of the treebank tagset (Xue et al., 2005). It consists of 17 unique labels.

The rules we extract belong to one of the following categories:

- γ contains only words, and ℓ is NULL; this corresponds to the general blocks used in phrase-based decoder (Och and Ney, 2004);
- γ contains words and variables of [X,0] and [X,1], and ℓ is NULL; this corresponds to the Hiero rules as in Chiang (2005);
- γ contains words and variables in the form of [X,TAG[2]], in which TAG is from the LDC tagset; this defines a well formed subtree, in which at least one child (constituent) is aligned to continuous target ngrams. If γ contains only variables from LDC tag set, this indicates all the constituents (children) in the subtree are aligned. This is a superset of rules generalizing those in Wang et al. (2007). If γ contains variables from POS tags, this essentially produces a superset of the monolingual side POS-based reordering rules explored in Tillmann (2008).

We focus on the third category — a syntactic label ℓ over the span of γ, indicating the covered source words consist of a linguistically well-defined phrase. ℓ together with γ define a tree-like structure: the root node is ℓ, and the aligned children are nonterminals in γ. The structure information is encoded in (ℓ, γ) pair-wise connections, and the variables keep the generalizations over atomic translation-pairs similar to Hiero models. When the rule is applied during decoding time, the labels, the tree-structure and the lexical items need to be all matched.

3 Learning and Applying Rules

A parser is assumed for the source language in the parallel data. In our case, a Chinese parser is applied for training and test data. A word alignment model is used to align the source words with the target words.

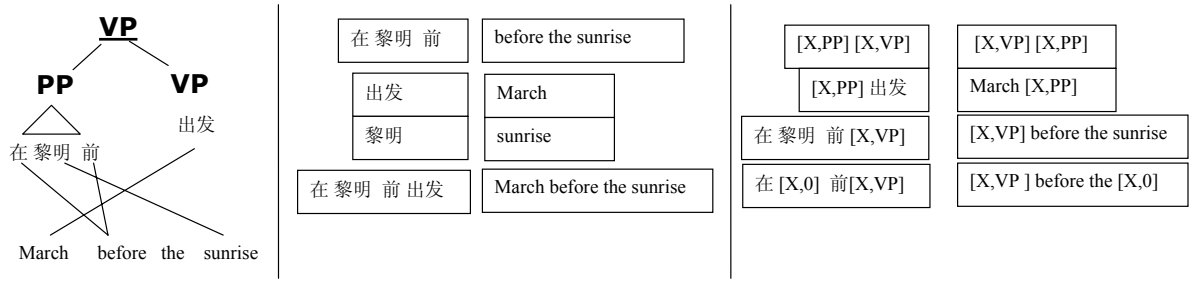
3.1 Extractions

Our rule extraction is a three-step process. First, traditional blocks (phrase-pairs) extraction is carried out. Secondly, Tree-to-String rules, are then extracted from the aligned blocks, of which the source side is covered by a complete subtree, with different permutations of the embedded aligned constituents, or partially lexicalized constituents. Otherwise, the Hiero-like rules will be extracted when there is no sub-tree structure identified, in our final step. Frequencies of extracted rules were counted to compute feature functions.

Figure 1-(a) shows that a subtree (with root at VP) is aligned to the English string. Considering the huge quantity of all the permutations of the aligned constituents under the tree, only part of the Tree-to-String rules extracted are shown in Figure 1-(c). The variables incorporate linguistic information in the assigned tag by the parser. When there is no aligned constituent for further generalization, the variables, defined in our grammar, back off to the Hiero-like ones without any label-identity information. One such example is in the rule "在 [X,0] 前 [X,VP] → [X,VP] before the [X,0]", in which the Hiero-style

[1] we use end-user terminologies for *source* and *target*.

[2] we index the tags for multiple occurrences in one rule



(a) Parse-Tree Alignment (b) Blocks Alignment (c) Tree-to-String rules with root of **VP**

Figure 1: Example rules extracted. (a) the aligned source parse tree with target string; (b) general blocks alignment; (c) Tree-to-String rules, with root of VP. The tree structure is aligned with target strings

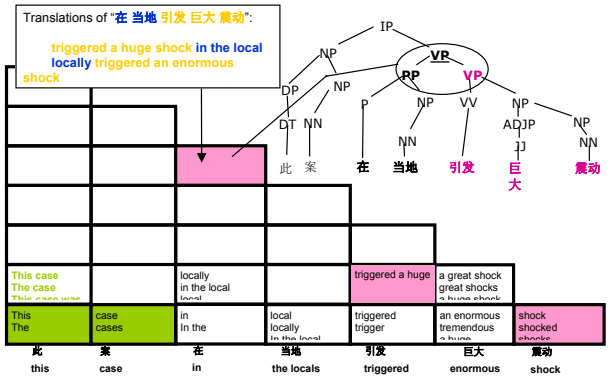


Figure 2: Subtree of "VP(PP,VP)" triggered a reordering pattern of swapping the order of the two children PP and VP in the source parse tree. This will move the translation "in the local" after the translation of "triggered a huge shock", to form the preferred translation in the highlighted cell: "triggered a huge shock in the local".

variable [X,0] and the label-based variable [X,VP] co-exist in our proposed rule.

We illustrate several special cases of our extracted Tree-to-String rules in the following. We index the variables with their positions to indicate the alignment $\sim$, and skip the feature function $\bar{w}$ to simplify the notations.

$$X \rightarrow < [X, IP]; [X, NP0]\ [X, VP0]; \quad (2)$$
$$[X, NP0]\ is\ [X, VP0] > .$$

The rule in Eqn. 2 shows that a source tree rooted at IP, with two children of NP and VP generalized into variables [X,NP] and [X,VP]; they are rewritten into "[X,NP] is [X,VP]", with the spontaneous word *is* inserted. Such rules are not allowed in Hiero-style models, as there is no lexical item between the two variables (Chiang, 2005) in the source side. This rule will generate a spontaneous word "is" from the given subtree structure. Usually, it is very hard to align the spontaneous word correctly, and the rules we proposed indicate that spontaneous words are generated directly from the source sub-tree structure, and they might not necessarily get aligned to some particular source words.

A second example is shown in Eqn. 3, which is similar to the Hiero rules:

$$X \rightarrow < \emptyset; [X, 0]\ \text{zhiyi}; \quad (3)$$
$$one\ of\ the\ [X, 0] > .$$

The rule in Eqn. 3 shows that when there is no linguistically-motivated root covering the span, ([X,NULL] is then assigned), we simply back off to the Hiero rules. In this case, the source span of $[X, 0]$ zhiyi is rewritten into the target "$one\ of\ the\ [X, 0]$", without considering the map-

ping of the root of the span. In this way, the representation power is kept in the variables in our rules, even if the source subtree is aligned to a discontinuous sequence on the target side. This is important for Chinese-to-English, because the grammar structure is so different that more than 40% of the subtree structures were not kept during the translation in our study on hand-aligned data. Following strictly the source side syntax will derail from these informative translation patterns.

$$X \quad \rightarrow< \quad [X, NP]; [X, NN1][X, NN2][X, NN3]; \\ [X, NN3][X, NN1][X, NN2] > . \qquad (4)$$

Eqn. 4. is a POS-based rule — a special case in our proposed rules. This rule shows the reordering patterns for three adjacent NN's. POS based rules can be very informative for some language-pairs such as Arabic-to-English, where the ADJ is usually moved before NN during the translations.

As also shown in Eqn. 4 for POS sequences, in the UPenn treebank-style parse trees, a root usually have more than two variables. Our rule set for subtree, therefore, contain more than two variables: "$X \rightarrow< [X, IP]; [X, ADVP0][X, NP0][X, VP0]; [X, NP0] [X, ADVP0][X, VP0] >$". A CYK-style decoder has to rely on *binarization* to preprocess the grammar as did in (Zhang et al., 2006) to handle multi-nonterminal rules. We adopt the so-called *dotted-rule* or *dotted-production*, similar to the Early-style algorithm (Earley, 1970), to handle the multi-nonterminal rules in our chart-based decoder.

3.2 Feature Functions

As used in most of the SMT decoders for a phrase-pair, a set of standard feature functions are applied in our decoder, including IBM Model-1 like scores in both directions, relative frequencies in both directions. In addition to these features, a counter is associated to each rule to collect how many rules were applied so far to generate a hypothesis. The standard Minimum Error Rate training (Och, 2003) was applied to tune the weights for all feature types.

The number of extracted rules from the GALE data is generally large. We pruned the rules according to their frequencies, and only keep at most the top-50 frequent candidates for each source side.

4 Chart-based Decoder

Given the source sentence, with constituent parse-trees, the decoder is to find the best derivation D^* which yield the English string e^*:

$$e^* = \arg\max_{D^*} \{\phi(D)\phi(e)\phi(f|e)\}, \qquad (5)$$

where $\phi(D)$ is the cost for each of the derivations that lead to e from a given source-parsed f; $\phi(e)$ is for cost functions from the standard n-gram language models; $\phi(f|e)$ is the cost for the standard translation models, including general blocks. We separate the costs for normal blocks and the generalized rules explicitly here, because the blocks contain stronger lexical evidences observed directly from data, and we assign them with less cost penalties via a different weight factor visible for optimization, and prefer the lexical match over the derived paths during the decoding.

Our decoder is a chart-based parser with beam-search for each cell in a chart. Because the tree-structure can have more than two children, therefore, the Tree-to-String rules extracted usually contain more than two variables. Slightly different from the decoder in (Chiang, 2005), we implemented the *dotted-rule* in Early-style parser to handle rules containing more than two variables. Our cube-expansion, implemented the cube-pruning in Chiang (2007), and integrated *piece-wise* cost computations for language models via LM states. The intermediate hypotheses were merged (recombined) according to their LM states and other cost model states. We use MER (Och, 2003) to tune the decoder's parameters using a development data set.

Figure 2 shows an example of a tree-based rule fired at the subtree of VP covering the highlighted cell. When a rule is applied at a certain cell in the chart, the covered source ngram should match not only the lexical items in the rules, but also the tree-structures as well. The two children under the subtree root VP are PP ("在当地": in the local) and VP ("引发巨大震动": triggered a huge shock). This rule triggered a swap of these children to generate the correct word order in the translation: "triggered a huge shock in the local".

5 Experiments

Our training data consists of two corpora: the GALE Chinese-English parallel corpus and the LDC hand-aligned corpus[1]. The Chinese side of these two corpora were parsed using a constituency parser (Luo, 2003). The average labeled F-measure of the parser is 81.4%.

Parallel sentences were first word-aligned using a MaxEnt aligner (Ittycheriah and Roukos, 2005). Then, phrase-pairs that overlap with our development and test set were extracted from the word alignments (from both hand alignments and automatically aligned GALE corpora) based on the projection principle (Tillmann, 2003). Besides the regular phrase-pairs, we also extracted the Tree-to-String rules from the two corpora. The detailed statistics are shown in Table 1. Our re-implementation of Hiero system is the baseline. We integrated the eleven reordering rules described in (Wang et al., 2007), in our chart-based decoder. In addition, we report the results of using the Tree-to-String rules extracted from the hand-aligned training data and the automatically aligned training data. We also report the result of our translation quality in terms of both BLEU (Papineni et al., 2002) and TER (Snover et al., 2006) against four human reference translations.

5.1 The Data

Table 1 shows the statistics of our training, development and test data. As our word aligner (Ittycheriah and Roukos, 2005) can introduce errors in extracting Tree-to-String rules, we use a small hand-aligned data set "CE16K", which consists of 16K sentence-pairs, to get relatively clean rules, free from alignment errors. A much larger GALE data set, which consists of 10 million sentence-pairs, is used to investigate the scalability of our proposed approach.

Table 1: Training and Test Data

Train/test	sentences	src words	tgt words
CE16K	16379	380103	477801
GALE	10.5M	274M	310M
MT03	919	24099	-
Dev07	2303	61881	-

[1]LDC2006E93

The NIST 2003 MT Evaluation (MT03) is used as our development data set to tune the decoder's parameters toward better BLEU score. The text part of GALE 2007 Chinese-to-English Development set (GALE DEV07) is used as our test set. MT03 consists of 919 sentences, whereas GALE DEV07 consists of 2303 sentences under two genres: NewsWire and WebLog. Both have four human reference translations.

5.2 Details of Extracted Rules

From the hand-aligned data, the rules we extracted fall into three categories: regular blocks (phrase-pairs), Hiero-like rules, and Tree-to-String rules. The statistics of the extracted rules are shown in Table 2

Table 2: Rules extracted from hand-aligned data

Types	Frequency
Block	846965
Hiero	508999
Tree-to-String	409767
Total	1765731

We focus on Tree-to-String rules. Table 3 shows the detailed statistics of the Tree-to-String rules extracted from the Chinese-to-English hand-aligned training data. The following section provides a detailed analysis of the most frequent subtrees observed in our training data.

5.2.1 Frequent Subtrees: NP, VP, and DNP

The majority of Tree-to-String rules we extracted are rooted at the following labels: NP (46%), VP(22.8%), DNP (2.23%), and QP(2.94%).

Wang et al. (2007) covers only subtrees of NP, VP, and LCP, which are a subset of our proposed Tree-to-String rules here. They apply these rules as a pre-processing step to reorder the input sentences with hard decisions. Our proposed Tree-to-String rules, on the contrary, are applied during the decoding process which allows for considering many possible competing reordering options for the given sentences, and the decoder will choose the best one according to the cost functions.

Table 4 shows the statistics of reordering rules for subtrees rooted at VP. The statistics suggest that

Table 5: Hiero, Tree-Based (eleven rules in Wang et al. (2007)), and Tree-to-String Rules with "DE"

Ruleset	Root	Src	Tgt	Frequency
Hiero	NULL	[X,0] 的 [X,1]	[X,0] 's [X,1]	347
	NULL	[X,0] 的 [X,1]	[X,1] of [X,0]	306
	NULL	[X,0] 的 [X,1]	[X,0] of [X,1]	174
Tree-Based	NP	DNP(NP) NP	NP DNP(NP)	-
	NP	DNP(PP) NP	NP DNP(PP)	-
	NP	DNP(LCP) NP	NP DNP(LCP)	-
Tree-to-String	[X,DNP]	[X,NP] [X,DEG]	[X,NP] [X,DEG]	580
	[X,DNP]	[X,NP] [X,DEG]	[X,DEG] [X,NP]	2163
	[X,DNP]	[X,NP] [X,DEG]	[X,NP] , [X,DEG]	4

Table 3: Distributions of the NP, VP, QP, LCP rules

Root	Frequency	Percentage (%)
NP	189616	46.2
VP	93535	22.8
IP	68341	16.6
PP	18519	4.51
DNP	9141	2.23
QP	12064	2.94
LCP	4127	1.00
CP	2994	0.73
PRN	2810	0.68
DP	1415	0.34
Others	6879	1.67
Total	409767	-

Table 4: Distribution of the reordering rules for subtrees rooted at VP: [X,VP]; [X,PP] [X,VP]; statistics are collected from GALE training data

Root	Target	Frequency
VP	[X,PP] [X,VP]	126310
	[X,VP] [X,PP]	22144
	[X,PP] , [X,VP]	1524
	[X,PP] that [X,VP]	1098
	[X,PP] and [X,VP]	831

it is impossible to come up with a reordering rule that is always applicable. For instance, (Wang et al., 2007) will always swap the children of the subtree VP(PP,VP). However, the statistics shown in Table 4 suggest that might not be best way. In fact, due to parser's performance and word alignment accuracies, the statistics we collected from the GALE dataset, containing 10 million sentence-pairs, show that the children in the subtree VP(PP,VP) is translated monotonically 126310 times, while reordered of only 22144 times. However, the hand-aligned data support the swap for 1245 times, and monotonically for only 168 times. Part of this disagreement is due to the word segmentation errors, incorrect word alignments and unreliable parsing results.

Another observations through our extracted Tree-to-String rules is on the controlled insertion of the target spontaneous[2] (function) words. Instead of hypothesizing spontaneous words based only on the language model or only on observing in phrase-pairs, we make use of the Tree-to-String rules to get suggestion on the insertion of spontaneous words. In this way, we can make sure that the spontaneous words are generated from the structure information, as opposed to those from a pure hypothesis. The advantage of this method is shown in Table 4. For instance, the word "that" and the punctuation "," were generated in the target side of the rule. This proves that our model can provide a more principled way to generate spontaneous words needed for fluent translations.

5.2.2 DEG and DEC

An interesting linguistic phenomenon that we investigated is the Chinese word DE "的". "的" is an informative lexical clue that indicates the need for long range phrasal movements. Table 5 shows a few

[2]Target spontaneous words are function words that do not have specific lexical source informants and are needed to make the target translation fluent.

high-frequent reordering rules that contain the Chinese word "DE".

The three type of rules handle "DE" differently. A major difference is the structure in the source side. Hiero rules do not consider any structure, and apply the rule of "[X,0] 的 [X,1]". Tree-based rules, as described in Wang et al. (2007) do not handle 的 directly; they are often implicitly taken care of when reordering DNPs instead. Our proposed Tree-to-String rules model 的 directly in a subtree containing DEG/DEC, which triggers word reordering within the structure. Our rule set includes all the above three rule-types with the associated frequencies, this enriched the reordering choices to be chosen by the chart-based decoder, guided by the statistics collected from the data and the language model costs.

5.3 Evaluation

We tuned the decoding parameters using the MT03 data set, and applied the updated parameters to the GALE evaluation set. The *eleven* rules of VP, NP, and LCP (tree-based) improved the Hiero baseline[3] from 32.43 to 33.02 on BLEU. The reason, the tree-reordering does not gain much over Hiero baseline, is probably that the reordering patterns covered by tree-reordering rules, are potentially handled in the standard Hiero grammar.

A small but noticeable further improvement over tree-based rules, from 33.02 to 33.26, was obtained on applying Tree-to-String rules extracted from hand-aligned dataset. We think that the Tree-based rules covers major reordering patterns for Chinese-English, and our hand-aligned dataset is also too small to capture representative statistics and more reordering patterns. A close check at the rules we learned from the hand-aligned data shows that the tree-based rules are simply the subset of the rules extracted. The Tree-to-String grammar improved the Hiero baseline from 32.43 to 33.26 on BLEU; considering the effects from the tree-based rules only, the additional information improved the BLEU scores from 33.02 to 33.26. Similar pictures of improvements were observed for the two unseen tests of newswire and weblog in GALE data.

When applying the rules extracted from the much larger GALE training set with about ten million sentence-pairs, we achieved significant improvements from both genres (newswire and web data). The improvements are significant in both BLEU and TER. BLEU improved from 32.44 to 33.51 on newswire, and from 25.88 to 27.91 on web data. Similar improvements were found in TER as shown in the table. The gain came mostly from the richer extracted rule set, which not only presents robust statistics for reordering patterns, but also offers more target spontaneous words generated from the syntactic structures. Since the top-frequent rules extracted are NP, VP, and IP as shown in Table 3, our proposed rules will be able to win the correct word order with reliable statistics, as long as the parser shows acceptable performances on these structures. This is especially important for weblog data, where the parser's overall accuracy potentially might not be very good.

Table 7 shows the translations from different grammars for the same source sentence. Both Tree-based and Tree-to-String methods get the correct reordering, while the latter can suggest insertions of target spontaneous words like "a" to allow the translation to run more fluently.

6 Conclusion and Discussions

In this paper, we proposed our approach to model both local and non-local word-reordering in one probabilistic synchronous CFG. Our current model incorporates source-side syntactic information, to model the observations that the source syntactic constituent tends to move together during translations. The proposed rule set generalizes over the variables in Hiero-rules, and we also showed the special cases of the Tree-based rules and the POS-based rules. Since the proposed rules has at most one-level tree structure, they can be easily applied in a chart-based decoder. We analyzed the statistics of our rules, qualitatively and quantitatively. Next, we compared our work with other research, especially with the work in Wang et al. (2007). Finally, we reported our empirical results on Chinese-English translations. Our Tree-to-String rules showed significant improvements over the Hiero baseline on the GALE DEV07 test set.

Given the low accuracy of the parsers, and the potential errors from Chinese word-segmentations, and

[3] Hiero results are from our own re-implementation.

Table 6: Hiero, Tree-Based (NP, VP, LCP), and Tree-to-String rules extracted from hand-aligned data (H) or from GALE training data (G)

Setup	MT03		GALE07-NewsWire		GALE07-Weblog	
	BLEUr4n4	TER	BLEUr4n4	TER	BLEUr4n4	TER
Hiero	32.43	59.75	31.68	61.45	25.99	65.65
Tree-based	33.02	59.84	32.22	61.46	25.67	65.64
Tree-to-String (H)	33.26	61.04	32.44	61.36	25.88	65.54
Tree-to-String (G)	35.51	57.28	33.51	59.71	27.91	62.88

Table 7: Hiero, Tree-Based (NP, VP, LCP), Tree-to-String Translations

Src-Sent	此案在当地引发巨大震动。
Hiero	in this case local triggered shock .
Tree-Based	the case triggered uproar in the local.
Tree-to-String	the case triggered a huge uproar in the local .

word-alignments, our rules learned are still noisy. Exploring better cost functions associate each rule might lead to further improvement. Because of the relative high accuracy of English parsers, many works such as Zollmann and Venugopal (2006) and Shen et al. (2008) emphasize on using syntax in target languages, to directly influence the fluency aspect of the translation output. In future, we plan to incorporate features from target-side syntactic information, and connect them with the source information explored in this paper, to model long-distance reordering for better translation quality.

Acknowledgments

The authors would like to thank the anonymous reviewers for their comments to improve this paper. This work was supported by DARPA GALE program under the contract number HR0011-06-2-0001.

References

Yaser Al-Onaizan and Papineni. Kishore. 2006. Distortion models for statistical machine translation. In *Proceedings of ACL-COLING*, pages 529–536.

David Chiang. 2005. A hierarchical phrase-based model for statistical machine translation. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL'05)*, pages 263–270, Ann Arbor, Michigan, June. Association for Computational Linguistics.

David Chiang. 2007. Hierarchical phrase-based translation. In *Computational Linguistics*.

Michael Collins, Philipp Koehn, and Ivona Kucerova. 2005. Clause restructuring for statistical machine translation. In *Proceedings of ACL*.

Jay Earley. 1970. An efficient context-free parsing algorithm. In *Communications of the ACM.*, volume 13, pages 94–102.

Heidi J. Fox. 2002. Phrasal cohesion and statistical machine translation. In *Proc. of the Conference on Empirical Methods in Natural Language Processing*, pages 304–311, Philadelphia, PA, July 6-7.

Jonathan Graehl and Kevin Knight. 2004. Training tree transducers. In *Proc. NAACL-HLT*.

Abraham Ittycheriah and Salim Roukos. 2005. A maximum entropy word aligner for arabic-english machine translation. In *HLT/EMNLP*.

Shankar Kumar and William Byrne. 2005. Local phrase reordering models for statistical machine translation. In *HLT/EMNLP 2005*, Vancouver, B.C., Canada.

Yang Liu, Qun Liu, and Shouxun Lin. 2006. Tree-to-string alignment template for statistical machine translation. In *ACL-Coling*.

Yang Liu, Yun Huang, Qun Liu, and Shouxun Lin. 2007. Forest-to-string statistical translation rules. In *45th Annual Meeting of the Association for Computational Linguistics*.

Xiaoqiang Luo. 2003. A maximum entropy chinese character-based parser. In *Proc. of ACL*.

Yuval Marton and Philip Resnik. 2008. Soft syntactic constraints for hierarchical phrased-based translation. In *ACL*.

Sonja Niessen and Hermann Ney. 2004. Statistical machine translation with scarce resources using morphosyntactic information. In *Computational Linguistics*.

Franz J. Och and Hermann Ney. 2004. The alignment template approach to statistical machine translation. In *Computational Linguistics*, volume 30, pages 417–449.

Franz J. Och. 2003. Minimum error rate training for statistical machine translation. In *Proc. of the 41st Annual Meeting of the Association for Computational Linguistics*, Japan, Sapporo, July.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proc. of the 40th Annual Conf. of the Association for Computational Linguistics (ACL 02)*, pages 311–318, Philadelphia, PA, July.

Libin Shen, Jinxi Xu, and Ralph Weischedel. 2008. A new string-to-dependency machine translation algorithm with a target dependency language model. In *Proceedings of ACL*.

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

Christoph Tillmann and Tong Zhang. 2005. A localized prediction model for statistical machine translation. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL'05)*, pages 557–564, Ann Arbor, Michigan, June. Association for Computational Linguistics.

Christoph Tillmann. 2003. A projection extension algorithm for statistical machine translation. In *Proc. of the Conference on Empirical Methods in Natural Language Processing*.

Christoph Tillmann. 2008. A rule-driven dynamic programming decoder for statistical mt. In *HLT Second Workshop on Syntax and Structure in Statistical Translation*.

Chao Wang, Michael Collins, and Phillip Koehn. 2007. Chinese syntactic reordering for statistical machine translation. In *proceedings of EMNLP*.

Dekai Wu. 1997. Stochastic inversion transduction grammars and bilingual parsing of parallel corpora. In *Computational Linguistics*, volume 23(3), pages 377–403.

Fei Xia and Michael McCord. 2004. Improving a statistical mt system with automatically learned rewrite patterns. In *the 20th International Conference on Computational Linguistics (COLING 2004)*, Geneva, Switzerland, Aug 22-29.

Deyi Xiong, Qun Liu, and Shouxun Lin. 2006. Maximum entropy based phrase reordering model for statistical machine translation. In *ACL-Coling*.

Nianwen Xue, Fei Xia, Fu-Dong Chiou, and Martha Palmer. 2005. The penn chinese treebank: Phrase structure annotation of a large corpus. In *Natural Language Engineering*, volume 11, pages 207–238.

K. Yamada and Kevin. Knight. 2001. Syntax-based Statistical Translation Model. In *Proceedings of the Conference of the Association for Computational Linguistics (ACL-2001)*.

Richard Zens, E. Matusov, and Hermmann Ney. 2004. Improved word alignment using a symmetric lexicon model. In *Proceedings of the 20th International Conference on Computational Linguistics (CoLing 2004)*, pages 36–42, Geneva, Switzerland, Auguest.

Hao Zhang, Liang Huang, Daniel Gildea, and Kevin Knight. 2006. Synchronous binarization for machine translation. In *Proceedings of the HLT-NAACL*.

Andreas Zollmann and Ashish Venugopal. 2006. Syntax augmented machine translation via chart parsing. In *Proc. of NAACL 2006 - Workshop on statistical machine translation*.