Two-Neighbor Orientation Model with Cross-Boundary Global Contexts

Hendra Setiawan, Bowen Zhou, Bing Xiang and Libin Shen
IBM T.J.Watson Research Center
1101 Kitchawan Road
Yorktown Heights, NY 10598, USA
{hendras,zhou,bxiang,lshen}@us.ibm.com

Abstract

Long distance reordering remains one of the greatest challenges in statistical machine translation research as the key contextual information may well be beyond the confine of translation units. In this paper, we propose Two-Neighbor Orientation (TNO) model that jointly models the orientation decisions between anchors and two neighboring multi-unit chunks which may cross phrase or rule boundaries. We explicitly model the longest span of such chunks, referred to as Maximal Orientation Span, to serve as a global parameter that constrains underlying local decisions. We integrate our proposed model into a state-of-the-art string-to-dependency translation system and demonstrate the efficacy of our proposal in a large-scale Chinese-to-English translation task. On NIST MT08 set, our most advanced model brings around +2.0 BLEU and -1.0 TER improvement.

1 Introduction

Long distance reordering remains one of the greatest challenges in Statistical Machine Translation (SMT) research. The challenge stems from the fact that an accurate reordering hinges upon the model's ability to make many local and global reordering decisions accurately. Often, such reordering decisions require contexts that span across multiple translation units.[1] Unfortunately, previous approaches fall short in capturing such cross-unit contextual information that could be critical in reordering. Specifically, the popular distortion or lexicalized reordering models in phrase-based SMT focus only on making good local prediction (i.e. predicting the orientation of immediate neighboring translation units), while translation rules in syntax-based SMT come with a strong context-free assumption, which model only the reordering within the confine of the rules. In this paper, we argue that reordering modeling would greatly benefit from richer cross-boundary contextual information

We introduce a reordering model that incorporates such contextual information, named the Two-Neighbor Orientation (TNO) model. We first identify *anchors* as regions in the source sentences around which ambiguous reordering patterns frequently occur and *chunks* as regions that are consistent with word alignment which may span multiple translation units at decoding time. Most notably, anchors and chunks in our model may not necessarily respect the boundaries of translation units. Then, we jointly model the orientations of chunks that immediately precede and follow the anchors (hence, the name "two-neighbor") along with the maximal span of these chunks, to which we refer as Maximal Orientation Span (MOS).

As we will elaborate further in next sections, our models provide a stronger mechanism to make more accurate global reordering decisions for the following reasons. First of all, we consider the orientation decisions on both sides of the anchors simultaneously, in contrast to existing works that only consider one-sided decisions. In this way, we hope to upgrade the unigram formulation of existing reordering models to a higher order formulation. Second of all, we capture the reordering of chunks that may cross translation units and may be composed of multiple units, in contrast to ex-

[1] We define translation units as phrases in phrase-based SMT, and as translation rules in syntax-based SMT.

Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics, pages 1264–1274,
Sofia, Bulgaria, August 4-9 2013.

isting works that focus on the reordering between individual translation units. In effect, MOS acts as a global reordering parameter that guides or constrains the underlying local reordering decisions.

To show the effectiveness of our model, we integrate our TNO model into a state-of-the-art syntax-based SMT system, which uses synchronous context-free grammar (SCFG) rules to jointly model reordering and lexical translation. The introduction of nonterminals in the SCFG rules provides some degree of generalization. However as mentioned earlier, the context-free assumption ingrained in the syntax-based formalism often limits the model's ability to influence global reordering decision that involves cross-boundary contexts. In integrating TNO, we hope to strengthen syntax-based system's ability to make more accurate global reordering decisions.

Our other contribution in this paper is a practical method for integrating the TNO model into syntax-based translations. The integration is non-trivial since the decoding of syntax-based SMT proceeds in a bottom-up fashion, while our model is more natural for top-down parsing, thus the model's full context sometimes is often available only at the latest stage of decoding. We implement an efficient shift-reduce algorithm that facilitates the accumulation of partial context in a bottom-up fashion, allowing our model to influence the translation process even in the absence of full context.

We show the efficacy of our proposal in a large-scale Chinese-to-English translation task where the introduction of our TNO model provides a significant gain over a state-of-the-art string-to-dependency SMT system (Shen et al., 2008) that we enhance with additional state-of-the-art features. Even though the experimental results carried out in this paper employ SCFG-based SMT systems, we would like to point out that our models is applicable to other systems including phrase-based SMT systems.

The rest of the paper is organized as follows. In Section 2, we introduce the formulation of our TNO model. In Section 3, we introduce and motivate the concept of Maximal Orientation Span. In Section 4, we introduce four variants of the TNO model with different model complexities. In Section 5, we describe the training procedure to estimate the parameters of our models. In Section 6, we describe our shift-reduce algorithm which integrates our proposed TNO model into syntax-based SMT. In Section 7, we describe our experiments and present our results. We wrap up with related work in Section 8 and conclusion in Section 9.

2 Two-Neighbor Orientation Model

Given an aligned sentence pair $\Theta = (F, E, \sim)$, let $\Delta(\Theta)$ be all possible chunks that can be extracted from Θ according to: [2]

$$\{(f_{j_1}^{j_2}/e_{i_1}^{i_2}) : \forall j_1 \leq j \leq j_2, \exists i : (j,i) \in \sim, i_i \leq i \leq i_2 \wedge \forall i_1 \leq i \leq i_2, \exists j : (j,i) \in \sim, j_i \leq j \leq j_2\}$$

Our Two-Neighbor Orientation model (TNO) designates $\mathcal{A} \subset \Delta(\Theta)$ as anchors and *jointly* models the orientation of chunks that appear *immediately* to the left and to the right of the anchors as well as the identities of these chunks. We define anchors as chunks, around which ambiguous reordering patterns frequently occur. Anchors can be learnt automatically from the training data or identified from the linguistic analysis of the source sentence. In our experiments, we use a simple heuristics based on part-of-speech tags which will be described in Section 7.

More concretely, given $\mathcal{A} \subset \Delta(\Theta)$, let $a = (f_{j_1}^{j_2}/e_{i_1}^{i_2}) \in \mathcal{A}$ be a particular anchor. Then, let $\mathcal{C}_L(a) \subset \Delta(\Theta)$ be a's left neighbors and let $\mathcal{C}_R(a) \subset \Delta(\Theta)$ be a's right neighbors, iff:

$$\forall C_L = (f_{j_3}^{j_4}/e_{i_3}^{i_4}) \in \mathcal{C}_L(a) : j_4 + 1 = j_1 \quad (1)$$

$$\forall C_R = (f_{j_5}^{j_6}/e_{i_5}^{i_6}) \in \mathcal{C}_R(a) : j_2 + 1 = j_5 \quad (2)$$

Given $\mathcal{C}_L(a)$ and $\mathcal{C}_R(a)$, let $C_L = (f_{j_3}^{j_4}/e_{i_3}^{i_4})$ and $C_R = (f_{j_5}^{j_6}/e_{i_5}^{i_6})$ be a particular pair of left and right neighbors of $a = (f_{j_1}^{j_2}/e_{i_1}^{i_2})$. Then, the orientation of C_L and C_R are $O_L(C_L, a)$ and $O_R(C_R, a)$ respectively and each may take one of the following four orientation values (similar to (Nagata et al., 2006)):

- **Monotone Adjacent** (*MA*), if $(i_4 + 1) = i_1$ for O_L and if $(i_2 + 1) = i_5$ for O_R
- **Reverse Adjacent** (*RA*), if $(i_2 + 1) = i_3$ for O_L and if $(i_6 + 1) = i_1$ for O_R
- **Monotone Gap** (*MG*), if $(i_4 + 1) < i_1$ for O_L and if $(i_2 + 1) < i_5$ for O_R

[2] We represent a chunk as a source and target phrase pair $(f_{j_1}^{j_2}/e_{i_1}^{i_2})$ where the subscript and the superscript indicate the starting and the ending indices as such $f_{j_1}^{j_2}$ denotes a source phrase that spans from j_1 to j_2.

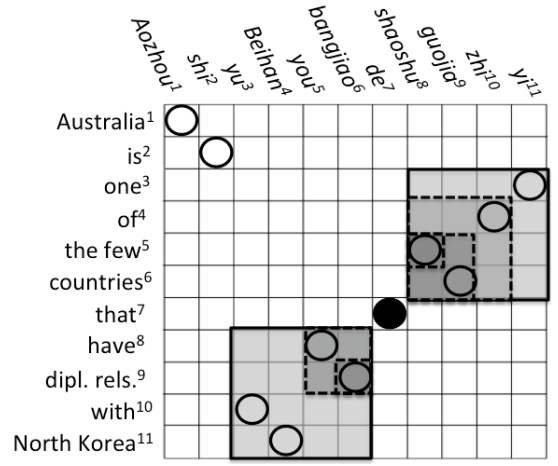

Figure 1: An aligned Chinese-English sentence pair. Circles represent alignment points. Black circle represents the anchor; boxes represent the anchor's neighbors.

- **Reverse Gap** (*RG*), if $(i_2 + 1) < i_3$ for O_L and if $(i_6 + 1) < i_1$ for O_R. (1)

The first clause (monotone, reverse) indicates whether the target order of the chunks follows the source order; the second (adjacent, gap) indicates whether the chunks are adjacent or separated by an intervening phrase when projected.

To be more concrete, let us consider an aligned sentence pair in Fig. 1, which is adapted from (Chiang, 2005). Suppose there is only one anchor, i.e. $a = (f_7^7/e_7^7)$ which corresponds to the word *de*(that). By applying Eqs. 1 and 2, we can infer that a has three left neighbors and four right neighbors, i.e. $\mathcal{C}_L(a) = (f_6^6/e_9^9), (f_5^6/e_8^9), (f_3^6/e_8^{11})$ and $\mathcal{C}_R(a) = (f_8^8/e_5^5), (f_8^9/e_5^6), (f_8^{10}/e_4^6), (f_8^{11}/e_3^6)$ respectively. Then, by applying Eq. 1, we can compute the orientation values of each of these neighbors, which are $O_L(\mathcal{C}_L(a), a) = RG, RA, RA$ and $O_R(\mathcal{C}_R(a), a) = RG, RA, RA, RA$. As shown, most of the neighbors have Reverse Adjacent (RA) orientation except for the smallest left and right neighbors (i.e. (f_6^6/e_9^9) and (f_8^8/e_5^5)) which have Reverse Gap (RG) orientation.

Given the anchors together with its neighboring chunks and their orientations, the Two-Neighbor Orientation model takes the following form:

$$\prod_{a \in \mathcal{A}} \sum_{\substack{C_L \in \mathcal{C}_L(a), \\ C_R \in \mathcal{C}_R(a)}} P_{TNO}(C_L, O_L, C_R, O_R | a; \Theta) \quad (2)$$

For conciseness, references that are clear from context, such the reference to C_L and a in $O_L(C_L, a)$, are dropped.

3 Maximal Orientation Span

As shown in Eq. 2, the TNO model has to enumerate all possible pairing of $C_L \in \mathcal{C}_L(a)$ and $C_R \in \mathcal{C}_R(a)$. To make the TNO model more tractable, we simplify the TNO model to consider only the largest left and right neighbors, referred to as the Maximal Orientation Span/MOS (M).

More formally, given $a = (f_{j_1}^{j_2}/e_{i_1}^{i_2})$, the left and the right MOS of a are:

$$M_L(a) = \underset{(f_{j_3}^{j_4}/e_{i_3}^{i_4}) \in \mathcal{C}_L(a)}{\arg\max} \; (j_4 - j_3)$$

$$M_R(a) = \underset{(f_{j_5}^{j_6}/e_{i_5}^{i_6}) \in \mathcal{C}_R(a)}{\arg\max} \; (j_6 - j_5)$$

Coming back to our example, the left and right MOS of the anchor are $M_L(a) = (f_3^6/e_8^{11})$ and $M_R(a) = (f_8^{11}/e_3^6)$. In Fig. 1, they are denoted as the largest boxes delineated by solid lines.

As such, we reformulate Eq. 2 into:

$$\prod_{a \in \mathcal{A}} \sum_{\substack{C_L \in \mathcal{C}_L(a), \\ C_R \in \mathcal{C}_R(a)}} P_{TNO}(M_L, O_L, M_R, O_R | a; \Theta).\delta_{\substack{C_L == M_L \wedge \\ C_R == M_R}} \quad (3)$$

where δ returns 1 if $(C_L == M_L \wedge C_R == M_R)$, otherwise 0.

Beyond simplifying the computation, the key benefit of modeling MOS is that it serves as a global parameter that can guide or constrain underlying local reorderings. As a case in point, let us consider a cheating exercise where we have to translate the Chinese sentence in Fig. 1 with the following set of hierarchical phrases[3]:

$X_a \rightarrow \langle$ *Aozhou*[1] *shi*[2] X_1, Australia[1] is[2] $X_1 \rangle$

$X_b \rightarrow \langle$ *yu*[3] *Beihan*[4] X_1, X_1 with[3] North[4] Korea$\rangle$

$X_c \rightarrow \langle$ *you*[5] *bangjiao*[6], have[5] dipl.[6] rels.$\rangle$

$X_d \rightarrow \langle X_1$ ***de***[7] *shaoshu*[8] *guojia*[9] *zhi*[10] *yi*[11], one[11] of[10] the few[8] countries[9] **that**[7] $X_1 \rangle$

This set of hierarchical phrases represents a translation model that has resolved all local ambiguities (i.e. local reordering and lexical mappings) except for the spans of the hierarchical phrases. With this example, we want to show that accurate local decisions (rather obviously) don't always lead to accurate global reordering and to demonstrate that explicit MOS modeling can play a crucial role to address this issue. To do so, we will again focus on the same anchor *de* (that).

[3] We use hierarchical phrase-based translation system as a case in point, but the merit is generalizable to other systems.

$\stackrel{d}{\Rightarrow}$ $\langle X_1$ ***de***^{7} *shaoshu*^{8} *guojia*^{9} *zhi*^{10} *yi*$^{11}\rangle$, $\langle$one^{11}of^{10}the few^8 countries^9 **that**$^7 X_1\rangle$

$\stackrel{a}{\Rightarrow}$ $\langle\langle$*Aozhou*^{1} *shi*^{2} $X_1\rangle$***de***^{7} *shaoshu*^{8} *guojia*^{9} *zhi*^{10} *yi*$^{11}\rangle$,
$\langle$one^{11}of^{10}the few^8 countries^9 **that**$^7\langle$Australia^1 is$^2 X_1\rangle\rangle$

$\stackrel{b}{\Rightarrow}$ $\langle\langle$*Aozhou*^{1} *shi*^{2} $\langle$*yu*^{3} *Beihan*^{4} $X_1\rangle\rangle$***de***^{7} *shaoshu*^{8} *guojia*^{9} *zhi*^{10} *yi*$^{11}\rangle$,
$\langle$one^{11}of^{10}the few^8 countries^9 **that**$^7\langle$Australia^1 is$^2\langle X_1$with^3 North^4 Korea$\rangle\rangle$

$\stackrel{c}{\Rightarrow}$ $\boxed{\langle_d}$ $\langle_a$*Aozhou*^{1} *shi*^{2} $\langle_b$*yu*^{3} *Beihan*^{4} $\langle_c$*you*^{5} *bangjiao*$^6\rangle_c\rangle_b\rangle_a$***de***^{7} *shaoshu*^{8} *guojia*^{9} *zhi*^{10} *yi*^{11} $\boxed{\rangle_d}$,
$\langle$one^{11}of^{10}the few^8 countries^9 **that**$^7\langle$Australia^1 is$^2\langle\langle$have^{5}dipl.^{6} rels.$\rangle$with^3 North^4 Korea$\rangle\rangle\rangle$

Table 1: Derivation of $X_d \prec X_a \prec X_b \prec X_c$ that leads to an incorrect translation.

$\stackrel{a}{\Rightarrow}$ $\langle$*Aozhou*^{1} *shi*$^2 X_1\rangle$, $\langle$Australia^1 is$^2 X_1\rangle$

$\stackrel{b}{\Rightarrow}$ $\langle$*Aozhou*^{1} *shi*$^2\langle$*yu*^{3}*Beihan*$^4 X_1\rangle\rangle$, $\langle$Australia^1 is$^2\langle X_1$with^3 North^4 Korea$\rangle\rangle$

$\stackrel{d}{\Rightarrow}$ $\langle$*Aozhou*^{1} *shi*$^2\langle$*yu*^{3}*Beihan*$^4\langle X_1$***de***^{7} *shaoshu*^{8} *guojia*^{9} *zhi*^{10} *yi*$^{11}\rangle\rangle\rangle$,
$\langle$Australia^1 is$^2\langle\langle$one^{11}of^{10}the few^8 countries^9 **that**$^7 X_1\rangle$with^3 North^4 Korea$\rangle\rangle$

$\stackrel{c}{\Rightarrow}$ $\langle_a$*Aozhou*^{1} *shi*$^2\langle_b$*yu*^{3} *Beihan*^{4} $\boxed{\langle_d}$ $\langle_c$*you*^{5} *bangjiao*$^6\rangle_c$***de***^{7} *shaoshu*^{8} *guojia*^{9} *zhi*^{10} *yi*^{11} $\boxed{\rangle_d}$ $\rangle_b\rangle_a$,
Australia^1 is$^2\langle\langle$one^{11}of^{10}the few^8 countries^9 **that**$^7\langle$have^{5}dipl.$^6\rangle\rangle$with^3 North^4 Korea$\rangle\rangle$

Table 2: Derivation of $X_a \prec X_b \prec X_d \prec X_c$ that leads to the correct translation.

As the rule's identifier, we attach an alphabet letter to each rule's left hand side, as such the anchor *de* (that) appears in rule X_d. We also attach the word indices as the superscript of the source words and project the indices to the target words aligned, as such "$have^5$" suggests that the word "have" is aligned to the 5-th source word, i.e. *you*. Note that to facilitate the projection, the rules must come with internal word alignment in practice. Now the indices on the target words in the rules are different from those in Fig. 1. We will also extensively use indices in this sense in the subsequent section about decoding. In such a sense, $M_L(a) = (f_3^6/e_3^6)$ and $M_R(a) = (f_8^{11}/e_8^{11})$.

Given the rule set, there are three possible derivations, i.e. $X_d \prec X_a \prec X_b \prec X_c$, $X_a \prec X_b \prec X_d \prec X_c$, and $X_a \prec X_d \prec X_b \prec X_c$, where $\prec$ indicates that the first operand dominates the second operand in the derivation tree. The application of the rules would show that the first derivation will produce an incorrect reordering while the last two will produce the correct ones. Here, we would like to point out that even in this simple example where all local decisions are made accurate, this ambiguity occurs and it would occur even more so in the real translation task where local decisions may be highly inaccurate.

Next, we will show that the MOS-related information can help to resolve this ambiguity, by focusing more closely on the first and the second derivations, which are detailed in Tables 1 and 2. Particularly, we want to show that the MOS generated by the incorrect derivation does not match the MOS learnt from Fig. 1. As shown, at the end of the derivation, we have all the information needed to compute the MOS (i.e. Θ) which is equivalent to that available at training time, i.e. the source sentence, the complete translation and the word alignment. Running the same MOS extraction procedure on both derivations would produce the right MOS that agrees with the right MOS previously learnt from Fig. 1, i.e. (f_8^{11}/e_8^{11}). However, that's not the case for left MOS, which we underline in Tables 1 and 2. As shown, the incorrect derivation produces a left MOS that spans six words, i.e. (f_1^6/e_1^6), while the correct derivation produces a left MOS that spans four words, i.e. (f_3^6/e_3^6). Clearly, the MOS of the incorrect derivation doesn't agree with the MOS we learnt from Fig. 1, unlike the MOS of the correct translation. This suggests that explicit MOS modeling would provide a mechanism for resolving crucial global reordering ambiguities that are beyond the ability of local models.

Additionally, this illustration also shows a case where MOS acts as a cross-boundary context which effectively relaxes the context-free assumption of hierarchical phrase-based formalism. In Tables 1 and 2's full derivations, we indicate rule boundaries explicitly by indexing the angle brackets, e.g. $\langle_a$ indicates the beginning of rule X_a in the derivation. As the anchor appears in X_d, we

highlight its boundaries in box frames. *de* (that)'s MOS respects rule boundaries if and only if all the words come entirely from X_d's antecedent or $\langle_d$ and $\rangle_d$ appears outside of MOS; otherwise it crosses the rule boundaries. As clearly shown in Table 2, the left MOS of the correct derivation (underlined) crosses the rule boundary (of X_d) since $\langle_d$ appears within the MOS.

Going back to the formulation, focusing on modeling MOS would simplify the formulation of TNO model from Eq. 2 into:

$$\prod_{a \in \mathcal{A}} P_{TNO}(M_L, O_L, M_R, O_R | a; \Theta) \quad (4)$$

which doesn't require enumerating of all possible pairs of $\mathcal{C}_L$ and $\mathcal{C}_R$.

4 Model Decomposition and Variants

To make the model more tractable, we decompose P_{TNO} in Eq. 4 into the following four factors: $P(M_R|a) \times P(O_R|M_R, a) \times P(M_L|O_R, M_R, a) \times P(O_L|M_L, O_R, M_R, a)$. Subsequently, we will refer to them as P_{M_R}, P_{O_R}, P_{M_L} and P_{O_L} respectively. Each of these factors will act as an additional feature in the log-linear framework of our SMT system. The above decomposition follows a generative story that starts from generating the right neighbor first. There are other equally credible alternatives, but based on empirical results, we settle with the above.

Next, we present four different variants of the model (not to be confused with the four factors above). Each variant has a different probabilistic conditioning of the factors. We start by making strong independence assumptions in Model 1 and then relax them as we progress to Model 4. The description of the models is as follow:

- **Model 1**. We assume P_{M_L} and P_{M_R} to be equal to 1 and $P_{O_R} \approx P(O_R|a; \Theta)$ to be independent of M_R and $P_{O_L} \approx P(O_L|a; \Theta)$ to be in independent of M_L, M_R and O_R.
- **Model 2**. On top of Model 1, we make P_{O_L} dependent on P_{O_R}, thus $P_{O_L} \approx P(O_L|O_R, a; \Theta)$.
- **Model 3**. On top of Model 2, we make P_{O_R} dependent on M_R and P_{O_L} on M_R and M_L, thus $P_{O_R} \approx P(O_R|M_R, a; \Theta)$ and $P_{O_L} \approx P(O_L|M_L, O_R, M_R; a, \Theta)$.
- **Model 4**. On top of Model 3, we model P_{M_R} and P_{M_L} as multinomial distributions estimated from training data.

Model 1 represents a model that focuses on making accurate one-sided decisions, independent of the decision on the other side. Model 2 is designed to address the deficiency of Model 1 since Model 1 may assign non-zero probability to improbable assignment of orientation values, e.g. Monotone Adjacent for the left neighbor and Reverse Adjacent for the right neighbor. Model 2 does so by conditioning P_{O_L} on O_R. In Model 3, we start incorporating MOS-related information in predicting O_L and O_R. In Model 4, we explicitly model the MOS of each anchor.

5 Training

The TNO model training consists of two different training regimes: 1) discriminative for training P_{O_L},P_{O_R}; and 2) generative for training P_{M_L}, P_{M_R}. Before describing the specifics, we start by describing the procedure to extract anchors and their corresponding MOS from training data, from which we collect statistics and extract features to train the model.

For each aligned sentence pair $(F, E, \sim)$ in the training data, the training starts with the identification of the regions in the source sentences as anchors ($\mathcal{A}$). For our Chinese-English experiments, we use a simple heuristic that equates as anchors, single-word chunks whose corresponding word class belongs to closed-word classes, bearing a close resemblance to (Setiawan et al., 2007). In total, we consider 21 part-of-speech tags; some of which are as follow: VC (copula), DEG, DEG, DER, DEV (*de*-related), PU (punctuation), AD (adjectives) and P (prepositions).

Next we generate all possible chunks $\Delta(\Theta)$ as previously described in Sec. 3. We then define a function $MinC(\Delta, j_1, j_2)$ which returns the shortest chunk that can span from j_1 to j_2. If $(f_{j_1}^{j_2}/e_{i_1}^{i_2}) \in \Delta$, then $MinC$ returns $(f_{j_1}^{j_2}/e_{i_1}^{i_2})$.

The algorithm to extract MOS takes Δ and an anchor $a = (f_{j_1}^{j_2}/e_{i_1}^{i_2})$ as input; and outputs the chunk that qualifies as MOS or none. Alg. 1 provides the algorithm to extract the right MOS; the algorithm to extract the left MOS is identical to Alg. 1, except that it scans for chunks to the left of the anchor. In Alg. 1, there are two intermediate parameters si and ei which represent the active search range and should initially be set to $j_2 + 1$ and $|F|$ respectively. Once we obtain $a, M_L(a)$ and $M_R(a)$, we compute $O_L(M_L(a), a)$ and $O_R(M_R(a), a)$ and are ready for training.

To estimate P_{O_L} and P_{O_R}, we train discriminative classifiers that predict the orientation values and use the normalized posteriors at decoding time as additional feature scores in SMT's log linear framework. We train the classifiers on a rich set of binary features ranging from lexical to part-of-speech (POS) and to syntactic features.

Algorithm 1: Function $M_R Ex$

input : $a = (f_{j_1}^{j_2}/e_{i_1}^{i_2})$, si, ei: int; Δ: chunks
output: $(f_{j_3}^{j_4}/e_{i_3}^{i_4})$: chunk or $\emptyset$
$(f_{j_3}^{j_4}/e_{i_3}^{i_4}) = MinC(\Delta, j_2 + 1, ei)$
if *($j_3 == j_2 + 1 \wedge j_4 == ei$)* **then**
  $\rightarrow f_{j_3}^{j_4}/e_{i_3}^{i_4}$
else
  if $(j_2 + 1 == ei)$ **then**
    $\rightarrow \emptyset$
  else
    if *(ei-2 $\leq$ si)* **then**
      $\rightarrow M_R Ex(a, si, ei - 1, \Delta)$
    else
      $m = \lceil$(si+ei)/2$\rceil$
      $(f_{j_3}^{j_4}/e_{i_4}^{i_4}) = MinC(\Delta, j_2 + 1, m)$
      if $(j_3 == j_2 + 1)$ **then**
        $c = M_R Ex(a, m, ei - 1, \Delta)$
        if *($c == \emptyset$)* **then**
          $\rightarrow f_{j_3}^{j_4}/e_{i_3}^{i_4}$
        else
          $\rightarrow c$
        end
      else
        $\rightarrow M_R Ex(a, si, m - 1, \Delta)$
      end
    end
  end
end

Suppose $a = (f_{j_1}^{j_2}/e_{i_1}^{i_2})$, $M_L(a) = (f_{j_3}^{j_4}/e_{i_3}^{i_4})$ and $M_L(a) = (f_{j_5}^{j_6}/e_{i_5}^{i_6})$, then based on the context's location, the elementary features employed in our classifiers can be categorized into:

1. *anchor-related*: `slex` (the actual word of $f_{j_1}^{j_2}$), `spos` (part-of-speech (POS) tag of `slex`), `sparent` (`spos`'s parent in the parse tree), `tlex` ($e_{i_1}^{i_2}$'s actual target word)..

2. *surrounding*: `lslex` (the previous word / $f_{j_1-1}^{j_1-1}$), `rslex` (the next word / $f_{j_2+1}^{j_2+1}$), `lspos` (`lslex`'s POS tag), `rspos` (`rslex`'s POS tag), `lsparent` (`lslex`'s parent), `rsparent` (`rslex`'s parent).

3. *non-local*: `lanchorslex` (the previous anchor's word) , `ranchorslex` (the next anchor's word), `lanchorspos` (`lanchorslex`'s POS tag), `ranchorspos` (`ranchorslex`'s POS tag).

4. *MOS-related*: `mosl_int_slex` (the actual word of $f_{j_3}^{j_3}$), `mosl_ext_slex` (the actual word of $f_{j_3}^{j_3}$), `mosl_int_spos` (`mosl_int_slex`'s POS tag), `mosl_ext_spos` (`mosl_ext_spos`'s POS tag), `mosr_int_slex` (the actual word of $f_{j_3}^{j_3}$), `mosr_ext_slex` (the actual word of $f_{j_3}^{j_3}$), `mosr_int_spos` (`mosr_int_slex`'s POS tag), `mosr_ext_spos` (`mosr_ext_spos`'s POS tag).

For Model 1, we train one classifier each for P_{O_R} and P_{O_L}. For Model 2-4, we train four classifiers for P_{O_L} for each value of O_R. We use only the MOS features for Model 3 and 4. Additionally, we augment the feature set with compound features, e.g. conjunction of the lexical of the anchor and the lexical of the left and the right anchors. Although they increase the number of features significantly, we found that these compound features are empirically beneficial.

We come up with > 50 types of features, which consist of a combination of elementary and compound features. In total, we generate hundreds of millions of such features from the training data. To keep the number features to a manageable size, we employ the L1-regularization in training to enforce sparse solutions, using the off-the-shelf LIBLINEAR toolkit (Fan et al., 2008). After training, the number of features in our classifiers decreases to below 5 million features for each classifier.

We train P_{M_L} and P_{M_R} via the relative frequency principle. To avoid the sparsity issue, we represent M_L as (`mosl_int_spos`,`mosl_ext_spos`) and M_R as (`mosr_int_spos`,`mosr_ext_spos`). We condition P_{M_L} and P_{M_R} only on `spos` and the orientation, estimating them as follow:

$$P(M_L|\texttt{spos}, O_L) = \frac{N(M_L, \texttt{spos}, O_L)}{N(\texttt{spos}, O_L)}$$
$$P(M_R|\texttt{spos}, O_R) = \frac{N(M_R, \texttt{spos}, O_R)}{N(\texttt{spos}, O_R)}$$

where N returns the count of the events in the training data.

		Target string (w/ source index)	Symbol(s) read	Op.	Stack(s)
(1)	X_c	have[5] dipl.[6] rels.	[5][6]	S,S,R	X_c:[5-6]
(2)	X_d	one[11] of[10] few[8] countries[9] **that**[7] X_c	[11][10]	S,S,R	[10-11]
(3)			[8][9]	S,S,R,R	[8-11]
(4)			[7]	S	[8-11]**[7]**
(5)			X_c:[5,6]	S	X_d:[8-11]**[7]**[5,6]
(6)	X_b	X_d with[3] North[4] Korea	X_d:[8-11]**[7]**[5,6]	S	[8-11]**[7]**[5,6]
(7)			[3][4]	S,S,R,R	X_b:[8-11]**[7]**[3-6]
(8)	X_a	Australia[1] is[2] X_b	[1][2]	S,S,R	[1-2]
(9)			X_b:[8-11]**[7]**[3,6]	S,A	X_a:[1-2][8-11]**[7]**[3,6]

Table 3: The application of the shift-reduce parsing algorithm, which corresponds to Table 2's derivation.

6 Decoding

Integrating the TNO Model into syntax-based SMT systems is non-trivial, especially with the MOS modeling. The method described in Sec. 3 assumes $\Theta = (F, E, \sim)$, thus it is only applicable at training or at the last stage of decoding. Since many reordering decisions may have been made at the earlier stages, the late application of TNO model would limit the utility of the model. In this section, we describe an algorithm that facilitates the incremental construction of MOS and the computation of TNO model on partial derivations.

The algorithm bears a close resemblance to the shift-reduce algorithm where a stack is used to accumulate (partial) information about a, M_L and M_R for each $a \in \mathcal{A}$ in the derivation. This algorithm takes an input stream and applies either the *shift* or the *reduce* operations starting from the beginning until the end of the stream. The *shift* operation advances the input stream by one symbol and push the symbol into the stack; while the *reduce* operation applies some reduction rule to the topmost elements of the stack. The algorithm terminates at the end of the input stream where the resulting stack will be propagated to the parent for the later stage of decoding. In our case, the input stream is the target string of the rule and the symbol is the corresponding source index of the elements of the target string. The reduction rule looks at two indices and merge them if they are adjacent (i.e. has no intervening phrase). We forbid the application of the reduction rule to anchors. Table 3 shows the execution trace of the algorithm for the derivation described in Table 2.

As shown, the algorithm starts with an empty stack. It then projects the source index to the corresponding target word and then enumerates the target string in a left to right fashion. If it finds a target word with a source index, it applies the shift operation, pushing the index to the stack. Unless the symbol corresponds to an anchor, it tries to apply the reduce operation. Line (4) indicates the special treatment to the anchor. If the symbol read is a nonterminal, then we push the entire stack that corresponds to that nonterminal. For example, when the algorithm reads X_d at line (6), it pushes the entire stack from line (5).

This algorithm facilitates the incremental construction of MOS which may cross rule boundaries. For example, at the end of the application of X_d at line (5), the current left MOS is [5-6]. However, the algorithm grows it to [3-6] after the application of rule X_b at line (7). Furthermore, it allows us to compute the models from partial hypothesis. For example, at line (5), we can compute P_{O_L} by considering [5,6] as M_L to be updated with [3,6] in line (7). This way, we expect our TNO model would play a bigger role at decoding time.

Specific to SCFG-based translation, the values of O_L and O_R are identical in the partial or in the full derivations. For example, the orientation values of *de* (that)'s left neighbor is always RA. This statement holds, even though at the end of Section 2, we stated that *de* (that)'s left neighbor may have other orientation values, i.e. RG for $C_L(a) = (f_6^6/e_9^9)$. The formal proof is omitted, but the intuition comes from the fact that the derivations for SCFG-based translation are subset of $\Delta(\Theta)$ and that (f_6^6/e_9^9) will never become M_L for $MinC(C_L(a), a)$ respectively (chunk that spans a and C_L). Consequently, for Model 1 and Model 2, we can obtain the model score earlier in the decoding process.

7 Experiments

Our baseline systems is a state-of-the-art string-to-dependency system (Shen et al., 2008). The system is trained on 10 million parallel sentences that are available to the Phase 1 of the DARPA BOLT Chinese-English MT task. The training corpora include a mixed genre of newswire, weblog, broadcast news, broadcast conversation, discussion forums and comes from various sources such as LDC, HK Law, HK Hansard and UN data.

In total, our baseline model employs about 40 features, including four from our proposed Two-Neighbor Orientation model. In addition to the standard features including the rule translation probabilities, we incorporate features that are found useful for developing a state-of-the-art baseline, such as the provenance features (Chiang et al., 2011). We use a large 6-gram language model, which was trained on 10 billion English words from multiple corpora, including the English side of our parallel corpus plus other corpora such as Gigaword (LDC2011T07) and Google News. We also train a class-based language model (Chen, 2009) on two million English sentences selected from the parallel corpus. As the backbone of our string-to-dependency system, we train 3-gram models for left and right dependencies and unigram for head using the target side of the bilingual training data. To train our Two-Neighbor Orientation model, we select a subset of 5 million aligned sentence pairs.

For the tuning and development sets, we set aside 1275 and 1239 sentences selected from LDC2010E30 corpus. We tune the decoding weights with PRO (Hopkins and May, 2011) to maximize BLEU-TER. As for the blind test set, we report the performance on the NIST MT08 evaluation set, which consists of 691 sentences from newswire and 666 sentences from weblog. We pick the weights that produce the highest development set scores to decode the test set.

Table 4 summarizes the experimental results on NIST MT08 newswire and weblog. In column 2, we report the classification accuracy on a subset of training data. Note that these numbers are for reference only and not directly comparable with each other since the features used in these classifiers include several gold standard information, such as the anchors' target words, the anchors' MOS-related features (Model 3 & 4) and the orientation of the right MOS (Model 2-4); all of which have to be predicted at decoding time.

	Acc	MT08 nw		MT08 wb	
		BLEU	TER	BLEU	TER
S2D	-	36.77	53.28	26.34	57.41
M1	72.5	37.60	52.70	27.59	56.33
M2	77.4	37.86	52.68	27.74	56.11
M3	84.5	38.02	52.42	28.22	**55.82**
M4	84.5	**38.55**	**52.41**	**28.44**	56.45

Table 4: The NIST MT08 results on newswire (nw) and weblog (wb) genres. S2D is the baseline string-to-dependency system (line 1), on top of which Two-Neighbor Orientation Model 1 to 4 are employed (line 2-5). The best TER and BLEU results on each genre are in **bold**. For BLEU, higher scores are better, while for TER, lower scores are better.

In columns 2 and 4, we report the BLEU scores, while in columns 3 and 5, we report the TER scores. The performance of our baseline string-to-dependency syntax-based SMT is shown in the first line, followed by the performance of our Two-Neighbor Orientation model starting from Model 1 to Model 4. As shown, the empirical results confirm our intuition that SMT can greatly benefit from reordering model that incorporate cross-unit contextual information.

Model 1 provides most of the gain across the two genres of around +0.9 to +1.2 BLEU and -0.5 to -1.1 TER. Model 2 which conditions P_{O_L} on O_R provides an additional +0.2 BLEU improvement on BLEU score consistently across the two genres. As shown in line 4, we see a stronger improvement in the inclusion of MOS-related information as features in Model 3. In newswire, Model 3 gives an additional +0.4 BLEU and -0.2 TER, while in weblog, it gives a stronger improvement of an additional +0.5 BLEU and -0.3 TER. The inclusion of explicit MOS modeling in Model 4 gives a significant BLEU score improvement of +0.5 but no TER improvement in newswire. In weblog, Model 4 gives a mixed results of +0.2 BLEU score improvement and a hit of +0.6 TER. We conjecture that the weblog text has a more ambiguous orientation span that are more challenging to learn. In total, our TNO model gives an encouraging result. Our most advanced model gives significant improvement of +1.8 BLEU/-0.8 TER in newswire domain and +2.1 BLEU/-1.0 TER over a strong string-to-dependency syntax-based SMT enhanced with additional state-of-the-art features.

8 Related Work

Our work intersects with existing work in many different respects. In this section, we mainly focus on work related to the probabilistic conditioning of our TNO model and the MOS modeling.

Our TNO model is closely related to the Unigram Orientation Model (UOM) (Tillman, 2004), which is the *de facto* reordering model of phrase-based SMT (Koehn et al., 2007). UOM views reordering as a process of generating (b, o) in a left-to-right fashion, where b is the current phrase pair and o is the orientation of b with the previously generated phrase pair b'. UOM makes strong independence assumptions and formulates the model as $P(o|b)$. Tillmann and Zhang (2007) proposed a Bigram Orientation Model (BOM) to include both phrase pairs (b and b') into the model. Their original intent is to model $P(o, b|o', b')$, but perhaps due to sparsity concerns, they settle with $P(o|b, b')$, dropping the conditioning on the previous orientation o'. Subsequent improvements use the $P(o|b, b')$ formula, for example, for incorporating various linguistics feature like part-of-speech (Zens and Ney, 2006), syntactic (Chang et al., 2009), dependency information (Bach et al., 2009) and predicate-argument structure (Xiong et al., 2012). Our TNO model is more faithful to the BOM's original formulation.

Our MOS concept is also closely related to hierarchical reordering model (Galley and Manning, 2008) in phrase-based decoding, which computes o of b with respect to a multi-block unit that may go beyond b'. They mainly use it to avoid overestimating "discontiguous" orientation but fall short in modeling the multi-block unit, perhaps due to data sparsity issue. Our MOS is also closely related to the efforts of modeling the span of hierarchical phrases in formally syntax-based SMT. Early works reward/penalize spans that respect the syntactic parse constituents of an input sentence (Chiang, 2005), and (Marton and Resnik, 2008). (Xiong et al., 2009) learn the boundaries from parsed and aligned training data, while (Xiong et al., 2010) learn the boundaries from aligned training data alone. Recent work couples span modeling tightly with reordering decisions, either by adding an additional feature for each hierarchical phrase (Chiang et al., 2008; Shen et al., 2009) or by refining the nonterminal label (Venugopal et al., 2009; Huang et al., 2010; Zollmann and Vogel, 2011). Common to this work is that the spans modeled may not correspond to MOS, which may be suboptimal as discussed in Sec. 3.

In equating anchors with the function word class, our work, particularly Model 1, is closely related to the function word-centered model of Setiawan et al. (2007) and Setiawan et al. (2009). However, we provide a discriminative treatment to the model to include a richer set of features including the MOS modeling. Our work in incorporating global context also intersects with existing work in Preordering Model (PM), e.g. (Niehues and Kolss, 2009; Costa-jussà and Fonollosa, 2006; Genzel, 2010; Visweswariah et al., 2011; Tromble and Eisner, 2009). The goal of PM is to reorder the input sentence F into F' whose order is closer to the target language order, whereas the goal of our model is to directly reorder F into the target language order. The crucial difference is that we have to integrate our model into SMT decoder, which is highly non-trivial.

9 Conclusion

We presented a novel approach to address a kind of long-distance reordering that requires global cross-boundary contextual information. Our approach, which we formulate as a Two-Neighbor Orientation model, includes the joint modeling of two orientation decisions and the modeling of the maximal span of the reordered chunks through the concept of Maximal Orientation Span. We describe four versions of the model and implement an algorithm to integrate our proposed model into a syntax-based SMT system. Empirical results confirm our intuition that incorporating cross-boundaries contextual information improves translation quality. In a large scale Chinese-to-English translation task, we achieve a significant improvement over a strong baseline. In the future, we hope to continue this line of research, perhaps by learning to identify anchors automatically from training data, incorporating a richer set of linguistics features such as dependency structure and strengthening the modeling of Maximal Orientation Span.

Acknowledgements

We would like to acknowledge the support of DARPA under Grant HR0011-12-C-0015 for funding part of this work. The views, opinions, and/or findings contained in this article/presentation are those of the author/ presenter and should not be interpreted as representing the official views or policies, either expressed or implied, of the DARPA.

References

Nguyen Bach, Qin Gao, and Stephan Vogel. 2009. Source-side dependency tree reordering models with subtree movements and constraints. In *Proceedings of the Twelfth Machine Translation Summit (MTSummit-XII)*, Ottawa, Canada, August. International Association for Machine Translation.

Pi-Chuan Chang, Huihsin Tseng, Dan Jurafsky, and Christopher D. Manning. 2009. Discriminative reordering with Chinese grammatical relations features. In *Proceedings of the Third Workshop on Syntax and Structure in Statistical Translation (SSST-3) at NAACL HLT 2009*, pages 51–59, Boulder, Colorado, June. Association for Computational Linguistics.

Stanley Chen. 2009. Shrinking exponential language models. In *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 468–476, Boulder, Colorado, June. Association for Computational Linguistics.

David Chiang, Yuval Marton, and Philip Resnik. 2008. Online large-margin training of syntactic and structural translation features. In *Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing*, pages 224–233, Honolulu, Hawaii, October.

David Chiang, Steve DeNeefe, and Michael Pust. 2011. Two easy improvements to lexical weighting. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 455–460, Portland, Oregon, USA, June. Association for Computational Linguistics.

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.

Marta R. Costa-jussà and José A. R. Fonollosa. 2006. Statistical machine reordering. In *Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing*, pages 70–76, Sydney, Australia, July. Association for Computational Linguistics.

Rong-En Fan, Kai-Wei Chang, Cho-Jui Hsieh, Xiang-Rui Wang, and Chih-Jen Lin. 2008. LIBLINEAR: A library for large linear classification. *Journal of Machine Learning Research*, 9:1871–1874.

Michel Galley and Christopher D. Manning. 2008. A simple and effective hierarchical phrase reordering model. In *Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing*, pages 848–856, Honolulu, Hawaii, October. Association for Computational Linguistics.

Dmitriy Genzel. 2010. Automatically learning source-side reordering rules for large scale machine translation. In *Proceedings of the 23rd International Conference on Computational Linguistics (Coling 2010)*, pages 376–384, Beijing, China, August. Coling 2010 Organizing Committee.

Mark Hopkins and Jonathan May. 2011. Tuning as ranking. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*, pages 1352–1362, Edinburgh, Scotland, UK., July. Association for Computational Linguistics.

Zhongqiang Huang, Martin Cmejrek, and Bowen Zhou. 2010. Soft syntactic constraints for hierarchical phrase-based translation using latent syntactic distributions. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, pages 138–147, Cambridge, MA, October. Association for Computational Linguistics.

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, June.

Yuval Marton and Philip Resnik. 2008. Soft syntactic constraints for hierarchical phrased-based translation. In *Proceedings of The 46th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 1003–1011, Columbus, Ohio, June.

Masaaki Nagata, Kuniko Saito, Kazuhide Yamamoto, and Kazuteru Ohashi. 2006. A clustered global phrase reordering model for statistical machine translation. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics*, pages 713–720, Sydney, Australia, July. Association for Computational Linguistics.

Jan Niehues and Muntsin Kolss. 2009. A POS-based model for long-range reorderings in SMT. In *Proceedings of the Fourth Workshop on Statistical Machine Translation*, pages 206–214, Athens, Greece, March. Association for Computational Linguistics.

Hendra Setiawan, Min-Yen Kan, and Haizhou Li. 2007. Ordering phrases with function words. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, pages 712–719, Prague, Czech Republic, June. Association for Computational Linguistics.

Hendra Setiawan, Min Yen Kan, Haizhou Li, and Philip Resnik. 2009. Topological ordering of function words in hierarchical phrase-based translation. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing*

of the AFNLP, pages 324–332, Suntec, Singapore, August. Association for Computational Linguistics.

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-08: HLT*, pages 577–585, Columbus, Ohio, June. Association for Computational Linguistics.

Libin Shen, Jinxi Xu, Bing Zhang, Spyros Matsoukas, and Ralph Weischedel. 2009. Effective use of linguistic and contextual information for statistical machine translation. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing*, pages 72–80, Singapore, August. Association for Computational Linguistics.

Christoph Tillman. 2004. A unigram orientation model for statistical machine translation. In *HLT-NAACL 2004: Short Papers*, pages 101–104, Boston, Massachusetts, USA, May 2 - May 7. Association for Computational Linguistics.

Christoph Tillmann and Tong Zhang. 2007. A block bigram prediction model for statistical machine translation. *ACM Transactions on Speech and Language Processing (TSLP)*, 4(3).

Roy Tromble and Jason Eisner. 2009. Learning linear ordering problems for better translation. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing*, pages 1007–1016, Singapore, August. Association for Computational Linguistics.

Ashish Venugopal, Andreas Zollmann, Noah A. Smith, and Stephan Vogel. 2009. Preference grammars: Softening syntactic constraints to improve statistical machine translation. In *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 236–244, Boulder, Colorado, June. Association for Computational Linguistics.

Karthik Visweswariah, Rajakrishnan Rajkumar, Ankur Gandhe, Ananthakrishnan Ramanathan, and Jiri Navratil. 2011. A word reordering model for improved machine translation. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*, pages 486–496, Edinburgh, Scotland, UK., July. Association for Computational Linguistics.

Deyi Xiong, Min Zhang, Aiti Aw, and Haizhou Li. 2009. A syntax-driven bracketing model for phrase-based translation. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP*, pages 315–323, Suntec, Singapore, August. Association for Computational Linguistics.

Deyi Xiong, Min Zhang, and Haizhou Li. 2010. Learning translation boundaries for phrase-based decoding. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 136–144, Los Angeles, California, June. Association for Computational Linguistics.

Deyi Xiong, Min Zhang, and Haizhou Li. 2012. Modeling the translation of predicate-argument structure for smt. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 902–911, Jeju Island, Korea, July. Association for Computational Linguistics.

Richard Zens and Hermann Ney. 2006. Discriminative reordering models for statistical machine translation. In *Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics (HLT-NAACL): Proceedings of the Workshop on Statistical Machine Translation*, pages 55–63, New York City, NY, June. Association for Computational Linguistics.

Andreas Zollmann and Stephan Vogel. 2011. A word-class approach to labeling pscfg rules for machine translation. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 1–11, Portland, Oregon, USA, June. Association for Computational Linguistics.