Discriminative Preordering Meets Kendall's τ Maximization

Sho Hoshino **Yusuke Miyao**
National Institute of Informatics / The Graduate University for Advanced Studies, Japan
{hoshino,yusuke}@nii.ac.jp

Katsuhito Sudoh **Katsuhiko Hayashi** **Masaaki Nagata**
NTT Communication Science Laboratories, NTT Corporation
{sudoh.katsuhito,hayashi.katsuhiko,nagata.masaaki}@lab.ntt.co.jp

Abstract

This paper explores a simple discriminative preordering model for statistical machine translation. Our model traverses binary constituent trees, and classifies whether children of each node should be reordered. The model itself is not extremely novel, but herein we introduce a new procedure to determine oracle labels so as to maximize Kendall's τ. Experiments in Japanese-to-English translation revealed that our simple method is comparable with, or superior to, state-of-the-art methods in translation accuracy.

1 Introduction

Current statistical machine translation systems suffer from major accuracy degradation in distant languages, primarily because they utilize exceptionally dissimilar word orders. One promising solution to this problem is *preordering*, in which source sentences are reordered to resemble the target language word orders, after which statistical machine translation is applied to reordered sentences (Xia and McCord, 2004; Collins et al., 2005). This is particularly effective for distant language pairs such as English and Japanese (Isozaki et al., 2010b).

Among such preordering, one of the simplest and straightforward model is a discriminative preordering model (Li et al., 2007), which classifies whether children of each constituent node should be reordered, given binary trees.[1] This simple model has, however, difficulty to find oracle labels. Yang et al. (2012) proposed a method to approximate oracle labels along dependency trees.

The present paper proposes a new procedure to find oracle labels. The main idea is simple: we determine reordering decisions in a way that maximizes Kendall's τ of word alignments. We prove that our procedure guarantees the optimal solution for word alignments given as an integer list; in a way that local decisions on each node reach global maximization of Kendall's τ in total. Any reordering methods that utilize word alignments along constituency benefit from this proof.

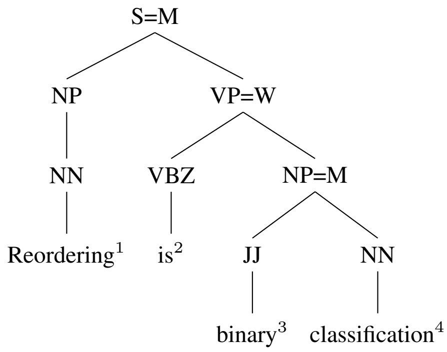

Figure 1: Discriminative preordering model.

Empirical study in Japanese-to-English translation demonstrate that our simple method outperforms a rule-based preordering method, and is comparable with, or superior to, state-of-the-art methods that rely on language-specific heuristics.

Our contributions are summarized as follows:

- We define a method for obtaining oracle labels in discriminative preordering as the maximization of Kendall's τ.
- We give a theoretical background to Kendall's τ based reordering for binary constituent trees.
- We achieve state-of-the-art accuracy in Japanese-to-English translation with a simple method without language-specific heuristics.

[1] It is also possible to use n-ary trees (Li et al., 2007; Yang et al., 2012), but we keep this binary model for simplicity.

Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Short Papers), pages 139–144, Beijing, China, July 26-31, 2015.

2 Preordering Method

2.1 Discriminative Preordering Model

The discriminative preordering model (Li et al., 2007) is a reordering model that determines whether children of each node should be reordered, given a binary constituent tree. For a sentence with n words, a node in a binary constituent tree is expressed as $v(i, p, j)$, where $1 \leq i \leq p < p + 1 \leq j \leq n$. This indicates that the node takes the left span from i-th to p-th words and the right span from $(p + 1)$-th to j-th words. Then we define whether a node should be reordered as $P(x \mid \theta(v(i, p, j)))$, where $x \in \{W, M\}$. W represents a reverse action (reorder the child nodes), M represents a monotonic action (do not reorder the child nodes), and θ is a feature function that is described at Section 2.4.

For instance, Figure 1 shows a sentence ($n = 4$) that has three binary nodes S, VP, and NP, which are our reordering candidates. We examine the NP node $v(3, 3, 4)$ that has a left ($binary^3$) and a right ($classification^4$) spans, of which reordering is determined by $P(x \mid \theta(v(3, 3, 4)))$, and is classified $x = M$ in this example. The actions for the VP node $v(2, 2, 4)$ and the S root node $v(1, 1, 4)$ are determined in a similar fashion.

Once all classifications are finished, the children of the nodes with W are reversed. From the constituent tree in Figure 1, this reordering produces a new tree in Figure 2 that represents a reordered sentence *Reordering binary classification is*, which is used in statistical machine translation.

2.2 Oracle Labels Maximizing Kendall's τ

In order to train such a classifier, we need an oracle label, W or M, for each node. Since we cannot rely on manual label annotation, we define a procedure to obtain oracle labels from word alignments. The principal idea is that we determine an oracle label of each node $v(i, p, j)$ so that it maximizes Kendall's τ under $v(i, p, j)$. This is intuitively a straightforward idea, because our objective is to find a monotonic order, which indicates maximization of Kendall's τ.

In the context of statistical machine translation, Kendall's τ is used as an evaluation metric for monotonicity of word orderings (Birch and Osborne, 2010; Isozaki et al., 2010a; Talbot et al., 2011). Given an integer list $\mathbf{x} = x_1, \ldots, x_n$, $\tau(\mathbf{x})$ measures a similarity between $\mathbf{x}$ and sorted $\mathbf{x}$ as:

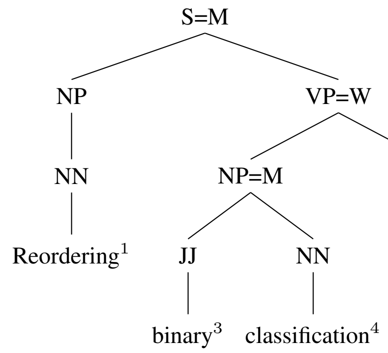

Figure 2: Output of discriminative preordering.

$$\tau(\mathbf{x}) = \frac{4c(\mathbf{x})}{n(n-1)} - 1,$$

where $c(\mathbf{x})$ is the number of concordant pairs between $\mathbf{x}$ and sorted $\mathbf{x}$, which is defined as:

$$c(\mathbf{x}) = \sum_{i,j \in [1,n], i<j} \delta(x_i < x_j),$$

where $\delta(x_i < x_j) = 1$ if $x_i < x_j$, and 0 otherwise. The τ function expresses that $\mathbf{x}$ is completely monotonic when $\tau(\mathbf{x}) = 1$, and in contrast, $\mathbf{x}$ is completely reversed when $\tau(\mathbf{x}) = -1$. Since $\tau(\mathbf{x})$ is proportional to $c(\mathbf{x})$, only $c(\mathbf{x})$ is considered in the course of our maximization.

Suppose that word alignments are given in the form $\mathbf{a} = a_1, \ldots, a_n$, where $a_x = y$ indicates that the x-th word in a source sentence corresponds to the y-th word in a target sentence.[2] We also assume that a binary constituent tree is given, and alignment for the span (i, j) is denoted as $\mathbf{a}(i, j)$. For each node $v(i, p, j)$, we define the score as:

$$\begin{aligned} s(v(i, p, j)) = {} & c(\mathbf{a}(i, p) \cdot \mathbf{a}(p + 1, j)) \\ & - c(\mathbf{a}(p + 1, j) \cdot \mathbf{a}(i, p)), \end{aligned}$$

where $\cdot$ indicates a concatenation of vectors. Then, a node that has $s(v(i, p, j)) < 0$ is assigned W, and a node that has $s(v(i, p, j)) > 0$ is assigned M. All the nodes scored as $s = 0$ are excluded from the training data, because they are noisy and ambiguous in terms of binary classification.

2.3 Proof of Independency over Constituency

The question then arises: *Can oracle labels achieve the best reordering in total?* We see this

[2] We used median values to approximate this y-th word in the target sentence for simplicity.

$t_{i:p}, t_{p+1:j}, w_{i:p}, w_{p+1:j}$,	$\sigma(v(i,p,j))$,
$t_{i:p} \circ t_{p+1:j}, w_{i:p} \circ w_{p+1:j}$,	$\sigma_r(v(i,p,j))$,
$t_{i:p} \circ t_{p+1:j} \circ w_{i:p} \circ w_{p+1:j}$,	$\sigma_t(v(i,p,j))$,
$t_{l:p}, t_{p+1:r}, w_{l:p}, w_{p+1:r}$,	$\sigma_w(v(i,p,j))$
$t_{l:p} \circ t_{p+1:r}, w_{l:p} \circ w_{p+1:r}$,	
$t_{l:p} \circ t_{p+1:r} \circ w_{l:p} \circ w_{p+1:r}$	

Table 1: Templates for the node $v(i,p,j)$: where integers l and r satisfy $i \leq l \leq p < p+1 \leq r \leq j$.

Template	Instance	Template	Instance
$t_{2:2}$	VBZ	$w_{2:2}$	is
$t_{3:4}$	JJ_NN	$w_{3:4}$	binary_classification
$t_{3:3}$	JJ	$w_{3:3}$	binary

Template	Instance
$\sigma(v(2,2,4))$	(VP(VBZis)(NP(JJbinary)(NNclassification)))
$\sigma_r(v(2,2,4))$	VP VBZ NP JJ NN VP_VBZ VP_NP NP_JJ NP_NN
$\sigma_t(v(2,2,4))$	(VP(VBZ)(NP(JJ)(NN)))
$\sigma_w(v(2,2,4))$	((is)((binary)(classification)))

Table 2: Examples in $v(2,2,4)$ from Figure 1.

Proposed	Accuracy	Previous	Accuracy
Full	**90.91**	Li et al. (2007)	84.43
w/o the first set	87.50		
w/o $\sigma(v(i,p,j))$	90.76		
w/o $\sigma_r(v(i,p,j))$	90.85		
w/o $\sigma_t(v(i,p,j))$	90.90		
w/o $\sigma_w(v(i,p,j))$	90.88		

Table 3: Ablation tests on binary classification accuracy (%).

is true, because $c(\mathbf{a}(i,j))$ can be computed in a recursive manner. See $c(\mathbf{a}(i,j))$ is decomposed as:

$$c(\mathbf{a}(i,j)) = c(\mathbf{a}(i,p)) + c(\mathbf{a}(p+1,j)) + \sum_{k\in[i,p],l\in[p+1,j]} \delta(a_k < a_l).$$

The three terms in this formula are mutually independent. That is, any reordering of $\mathbf{a}(i,p)$ changes only the first term and the others are unchanged. We maximize $c(\mathbf{a}(i,j))$ by maximizing each term. Since the first and the second terms are maximized recursively, our method directly maximizes the third term, which corresponds to our oracle labels, hence $c(\mathbf{a})$ and $\tau(\mathbf{a})$ of entire sentence.[3]

Essentially, our decisions on each node are equivalent to sorting a list consists of left and right points, while the order of the points inside of left and right lists are left untouched. We determine oracle labels for a given constituent tree by computing $s(v(i,p,j))$ for every $v(i,p,j)$ independently.

[3]Oracle labels guarantee $\tau(\mathbf{a}) \geq 0$, but not $\tau(\mathbf{a}) = 1$, because parsed trees will not correspond to word alignments.

		test9		test10	
Settings	DL	RIBES	BLEU	RIBES	BLEU
Baseline w/o preordering					
Moses	0	66.95	26.36	67.50	27.17
Moses	10	68.95	29.41	69.64	30.20
Moses	20	69.88	30.12	70.22	30.51
Proposed preordering					
Giza	0	77.49	33.08	77.49	33.65
Giza	10	77.44	**33.28**	77.42	33.77
Nile	0	77.74	32.97	77.89	**33.91**
Nile	10	**77.97**	**33.55**	**78.07**	**34.13**

Table 4: Results in Japanese-to-English translation. Boldfaces denote the highest scores and the insignificant difference ($p < 0.01$) from the highest scores in bootstrap resampling (Koehn, 2004).

2.4 Features

Table 1 shows the templates for the node $v(i,p,j)$ of the feature function θ in Section 2.1. To tell the differences between the left span $\mathbf{a}(i,p)$ and the right span $\mathbf{a}(p+1,j)$, such as whether the head word of the node is in left or right, the first set of templates considers individual indices *x:y* that denote the span from x-th to y-th words: where t_x represents a part-of-speech feature; w_x represents a lexical feature; and $\circ$ represents feature combination. The second set of templates considers constituent structures of the node by supplying three S-expressions and parent-child relations: where $\sigma(v(i,p,j))$ represents a constituent structure under the node $v(i,p,j)$; $\sigma_r(v(i,p,j))$ represents part-of-speech tags of the node and their parent-child relations; $\sigma_t(v(i,p,j))$ represents the constituent structure including only part-of-speech tags; and $\sigma_w(v(i,p,j))$ represents the constituent structure including only surface words.

Table 2 shows instances of features for the VP node $v(2,2,4)$ in Figure 1, which has the left (is^2) and the right ($binary^3\ classification^4$) spans.

Table 3 shows ablation test results on binary classification, which indicate that our templates performed better than that of Li et al. (2007).

3 Experiment

3.1 Experimental Settings

We perform experiments over the NTCIR patent corpus (Goto et al., 2011) that consists of more than 3 million sentences in English and Japanese. Following conventional literature settings (Goto et al., 2012; Hayashi et al., 2013), we used all 3 million sentences from the NTCIR-7 and NTCIR-

		test9				test10			
Reordering Methods	DL	RIBES	Δ	BLEU	Δ	RIBES	Δ	BLEU	Δ
Moses	20	69.88		30.12		70.22		30.51	
Proposed preordering	10	**77.97**	**+8.09**	**33.55**	**+3.43**	**78.07**	+7.85	**34.13**	**+3.62**
Moses (Hoshino et al., 2013)	20	68.08		27.57					
Preordering (Hoshino et al., 2013)	10	72.37	+4.29	30.56	+2.99				
Moses (Goto et al., 2012)	20	68.28		30.20					
Moses-chart (Goto et al., 2012)		70.64	+2.36	30.69	+0.49				
Postordering (Goto et al., 2012)		75.48	+7.20	33.04	+2.84				
Moses (Hayashi et al., 2013)	20	69.31		29.43		68.90		29.99	
Postordering (Hayashi et al., 2013)	0	76.46	+7.15	32.59	+3.16	76.76	**+7.86**	33.14	+3.15

Table 5: Comparison with previous systems in Japanese-to-English translation, of which scores are retrieved from their papers. Boldfaces indicate the highest scores and differences.

8 training sets, used the first 1000 sentences in NTCIR-8 development set, and then fetched both the NTCIR-9 and NTCIR-10 testing sets. The machine translation experiments pipelined Moses 3 (Koehn et al., 2007) with lexicalized reordering, SRILM 1.7.0 (Stolcke et al., 2011) in 6-gram order, MGIZA (Gao and Vogel, 2008), and RIBES (Isozaki et al., 2010a) and BLEU (Papineni et al., 2002) for evaluation. Binary constituent parsing in Japanese used Haruniwa (Fang et al., 2014), Berkeley parser 1.7 (Petrov and Klein, 2007), Comainu 0.7.0 (Kozawa et al., 2014), MeCab 0.996 (Kudo et al., 2004), and Unidic 2.1.2.

We explore two types of word alignment data for training our preordering model. The first data (*Giza*) is created by running an unsupervised aligner Giza (Och and Ney, 2003) on the training data (3 million sentences). The second data (*Nile*) is developed by training a supervised aligner Nile (Riesa et al., 2011) with manually annotated 8,000 sentences, then applied the trained alignment model to remaining training data. In the evaluation on manually annotated 1,000 sentences[4], Giza achieved F1 50.1 score, while Nile achieved F1 86.9 score, for word alignment task.

3.2 Result

Table 4 shows the performance of our method, which indicates that our preordering significantly improved translation accuracy in both RIBES and BLEU scores, from the baseline result attained by Moses without preordering. In particular, the preordering model trained with the Giza data revealed a substantial improvement, while the use of the Nile data further improves accuracy. This suggests that our method is particularly effective when high-accuracy word alignments are given. In addition, we achieved modest improvements even with *DL=0* (no distortion allowed), which indicates the monotonicity of our reordered sentences.

Table 5 shows a comparison of the proposed method with a rule-based preordering method (Hoshino et al., 2013) and two postordering methods (Goto et al., 2012; Hayashi et al., 2013).[5] One complication is that each work reports different baseline accuracy, although Moses is shared as a baseline, because these systems differ in various settings in data preprocessing, tokenization criteria, etc. Since this makes a fair comparison difficult, we additionally put a score difference (Δ) of each system from its own baseline.

Our proposed method showed translation accuracy comparable with, or superior to, state-of-the-art methods. This highlights the importance of Kendall's τ maximization in the simple discriminative preordering model. In contrast to a substantial gain in RIBES, we attained a rather comparable gain in BLEU. The investigation of our translation suggests that insufficient generation of English articles caused a significant degradation in the BLEU score. Previous systems listed in Table 5 incorporated article generation and demonstrated its positive effect (Goto et al., 2012; Hayashi et al., 2013). While we achieved state-of-the-art accuracy without language-specific techniques, it is also a promising direction to integrate our preordering method with language-specific techniques such as article generation and subject generation (Kudo et al., 2014).

[4]This testing data is excluded from latter experiments.

[5]We could not find a comparable report using tree-based machine translation systems apart from Moses-chart; nevertheless, Neubig and Duh (2014) reported that their forest-to-string system on the same corpus, which is unfortunately evaluated on the different testing data (test7), showed RIBES +6.19 (75.94) and BLEU +2.93 (33.70) improvements. Although not directly comparable, our method achieves a comparable or superior improvement.

4 Related Work

Li et al. (2007) proposed a simple discriminative preordering model as described in Section 2.1. They employed heuristics that utilize Giza to align their training sentences, then sort source words to resemble target word indices. After that, sorted source sentences without overlaps are used to train the model. They gained BLEU +1.54 improvement in Chinese-to-English evaluation. Our proposal follows their model, while we do not rely on their heuristics for preparing training data.

Lerner and Petrov (2013) proposed another discriminative preordering model along dependency trees, which classifies whether the parent of each node should be the head in target language. They reported BLEU +3.7 improvement in English-to-Japanese translation. Hoshino et al. (2013) proposed a similar but rule-based method for Japanese-to-English dependency preordering.

Yang et al. (2012) proposed a method to produce oracle reordering in the discriminative preordering model along dependency trees. Their idea behind is to minimize word alignment crossing recursively, which is essentially the same reordering objective as our Kendall's τ maximization. Since they targeted complex n-ary dependency instead of simple binary trees, their method only calculates approximated oracle reordering in practice by ranking principle. We did not take n-ary trees into consideration to follow the simple discriminative preordering model along constituency, while the use of binary trees enabled us to produce strict oracle reordering as a side effect.

Another research direction called postordering (Sudoh et al., 2011; Goto et al., 2012; Hayashi et al., 2013) has been explored in Japanese-to-English translation. They first translate Japanese input into head final English texts obtained by the method of Isozaki et al. (2010b), then reorder head final English texts into English word orders.

5 Conclusion

We proposed a simple procedure to train a discriminative preordering model. The main idea is to obtain oracle labels for each node by maximizing Kendall's τ of word alignments. Experiments in Japanese-to-English translation demonstrated that our procedure, without language-specific heuristics, achieved state-of-the-art translation accuracy.

Acknowledgments

We would like to thank Kevin Duh, Atsushi Fujita, Taku Kudo, Shinsuke Mori, Toshiaki Nakazawa, Graham Neubig, Hiroshi Noji, and anonymous reviewers for their insightful comments.

References

Alexandra Birch and Miles Osborne. 2010. LRscore for evaluating lexical and reordering quality in MT. In *Proceedings of the Joint Fifth Workshop on Statistical Machine Translation and MetricsMATR*, pages 327–332.

Michael Collins, Philipp Koehn, and Ivona Kucerova. 2005. Clause restructuring for statistical machine translation. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics*, pages 531–540.

Tsaiwei Fang, Alastair Butler, and Kei Yoshimoto. 2014. Parsing Japanese with a PCFG treebank grammar. In *Proceedings of the Twentieth Meeting of the Association for Natural Language Processing*, pages 432–435.

Qin Gao and Stephan Vogel. 2008. Parallel implementations of word alignment tool. In *Software Engineering, Testing, and Quality Assurance for Natural Language Processing*, pages 49–57.

Isao Goto, Bin Lu, Ka Po Chow, Eiichiro Sumita, and Benjamin K. Tsou. 2011. Overview of the patent machine translation task at the NTCIR-9 workshop. In *Proceedings of the NTCIR-9 Workshop Meeting*, pages 559–578.

Isao Goto, Masao Utiyama, and Eiichiro Sumita. 2012. Post-ordering by parsing for Japanese-English statistical machine translation. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics*, pages 311–316.

Katsuhiko Hayashi, Katsuhito Sudoh, Hajime Tsukada, Jun Suzuki, and Masaaki Nagata. 2013. Shift-reduce word reordering for machine translation. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1382–1386.

Sho Hoshino, Yusuke Miyao, Katsuhito Sudoh, and Masaaki Nagata. 2013. Two-stage pre-ordering for Japanese-to-English statistical machine translation. In *Proceedings of the Sixth International Joint Conference on Natural Language Processing*, pages 1062–1066.

Hideki Isozaki, Tsutomu Hirao, Kevin Duh, Katsuhito Sudoh, and Hajime Tsukada. 2010a. Automatic evaluation of translation quality for distant language pairs. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, pages 944–952.

Hideki Isozaki, Katsuhito Sudoh, Hajime Tsukada, and Kevin Duh. 2010b. Head finalization: A simple reordering rule for SOV languages. In *Proceedings of the Joint Fifth Workshop on Statistical Machine Translation and MetricsMATR*, pages 244–251.

Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, Chris Dyer, Ondrej Bojar, Alexandra Constantin, and Evan Herbst. 2007. Moses: Open source toolkit for statistical machine translation. In *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics*, pages 177–180.

Philipp Koehn. 2004. Statistical significance tests for machine translation evaluation. In *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing*, pages 388–395.

Shunsuke Kozawa, Kiyotaka Uchimoto, and Yasuharu Den. 2014. Adaptation of long-unit-word analysis system to different part-of-speech tagset (in Japanese). *Journal of Natural Language Processing*, 21(2):379–401.

Taku Kudo, Kaoru Yamamoto, and Yuji Matsumoto. 2004. Applying conditional random fields to japanese morphological analysis. In *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing*, pages 230–237.

Taku Kudo, Hiroshi Ichikawa, and Hideto Kazawa. 2014. A joint inference of deep case analysis and zero subject generation for Japanese-to-English statistical machine translation. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics*, pages 557–562.

Uri Lerner and Slav Petrov. 2013. Source-side classifier preordering for machine translation. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 513–523.

Chi-Ho Li, Minghui Li, Dongdong Zhang, Mu Li, Ming Zhou, and Yi Guan. 2007. A probabilistic approach to syntax-based reordering for statistical machine translation. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, pages 720–727.

Graham Neubig and Kevin Duh. 2014. On the elements of an accurate tree-to-string machine translation system. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics*, pages 143–149.

Franz Josef Och and Hermann Ney. 2003. A systematic comparison of various statistical alignment models. *Computational Linguistics*, 29(1):19–51.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. BLEU: a method for automatic evaluation of machine translation. In *Proceedings of 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318.

Slav Petrov and Dan Klein. 2007. Improved inference for unlexicalized parsing. In *Human Language Technologies 2007: The Conference of the North American Chapter of the Association for Computational Linguistics; Proceedings of the Main Conference*, pages 404–411.

Jason Riesa, Ann Irvine, and Daniel Marcu. 2011. Feature-rich language-independent syntax-based alignment for statistical machine translation. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*, pages 497–507.

Andreas Stolcke, Jing Zheng, Wen Wang, and Victor Abrash. 2011. SRILM at sixteen: Update and outlook. In *Proceedings of the IEEE Automatic Speech Recognition and Understanding Workshop*.

Katsuhito Sudoh, Xianchao Wu, Kevin Duh, Hajime Tsukada, and Masaaki Nagata. 2011. Post-ordering in statistical machine translation. In *Proceedings of the Machine Translation Summit XIII*, pages 316–323.

David Talbot, Hideto Kazawa, Hiroshi Ichikawa, Jason Katz-Brown, Masakazu Seno, and Franz Och. 2011. A lightweight evaluation framework for machine translation reordering. In *Proceedings of the Sixth Workshop on Statistical Machine Translation*, pages 12–21.

Fei Xia and Michael McCord. 2004. Improving a statistical MT system with automatically learned rewrite patterns. In *Proceedings of the 20th International Conference on Computational Linguistics*, pages 508–514.

Nan Yang, Mu Li, Dongdong Zhang, and Nenghai Yu. 2012. A ranking-based approach to word reordering for statistical machine translation. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics*, pages 912–920.