The ILLC-UvA SMT System for IWSLT 2010

Maxim Khalilov and Khalil Sima'an

Institute for Logic, Language and Computation
University of Amsterdam
Amsterdam, The Netherlands
{m.khalilov, k.simaan}@uva.nl

Abstract

In this paper we give an overview of the ILLC-UvA(Institute for Logic, Language and Computation - University of Amsterdam) submission to the 7th International Workshop on Spoken Language Translation evaluation campaign. It outlines the architecture and configuration of the novel feature we are introducing: a syntax-based model for source-side reordering via tree transduction.

We have concentrated on the Chinese-to-English and English-to-Chinese DIALOG translation tasks.

1. Introduction

This is the first time that ILLC-UvA participates in the IWSLT evaluation campaigns. In this paper, we describe the 2010 system's architecture describing the distinguishing features of our source permutation reordering model and issues of its adaptation to the task of speech translation.

We exploit the idea of augmenting statistical machine translation (SMT) by using a reordering step prior to translation that has proved to be successful in improving translation quality [1, 2].

Our system consists of two independent steps. First, we reorder the words of a sentence of the source language s with respect to the word order of the target language and a given source-side parse tree. Second, the reordered source sentence s' is monotonically translated into a target sentence t by a standard phrase-based SMT system. The discriminative reordering model based on syntax which is the core of our system was presented in [3] and is summarized next.

Figure 1 depicts the translation from source string s to target string t with alignment a (solid line) and the alternative of source reordering s into $s^{'}$ followed by the translation $s^{'} \rightarrow t$ with alignment $a^{'}$ (in dashed lines).

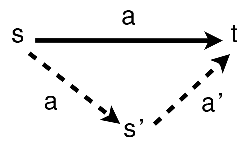

Figure 1: *Translation schemes with and without a reordering step.*

We define source permutation as the problem of learning how to *transfer* a given source parse-tree into a parse-tree that minimizes the divergence from target word-order. We model the tree transfer $\tau_s \rightarrow \tau_{s'}$ as a sequence of local, independent transduction operations, each transforming the current intermediate tree $\tau_{s^{'}_i}$ into the next intermediate tree $\tau_{s^{'}_{i+1}}$, with $\tau_{s_0} = \tau_s$ and $\tau_{s^{'}_n} = \tau_{s'}$. A transduction operation merely permutes the sequence of $m > 1$ children of a single node in an intermediate tree, i.e., unlike previous work, we do not binarize the trees. The number of permutations is factorial in m, and learning a sequence of transductions for explaining a source permutation can be computationally rather challenging (see [4]). Yet, from the limited perspective of source string permutation ($s \rightarrow s^{'}$), another challenge is to integrate a figure of merit that measures in how far $s^{'}$ resembles a plausible target word-order.

We contribute solutions to these challenging problems. Firstly, we learn the transduction operations using a discriminative estimate of $P(\pi(\alpha_x) \mid N_x, \alpha_x, context_x)$, where $\pi(\alpha_x)$ is a permutation of α_x (the ordered sequence of node labels under x), N_x is the label of node (address) x, $N_x \rightarrow \alpha_x$ is the context-free production under x, $\pi(\alpha_x)$ is a permutation of α_x and $context_x$ represents a surrounding syntactic context. As a result, this constrains $\{\pi(\alpha_x)\}$ only to those found in the training data, and it conditions the transduction application probability on its specific contexts. Secondly, in every sequence $s^{'}_0 = s, \ldots, s^{'}_n = s^{'}$ resulting from tree transductions, we prefer those local transductions on $\tau_{s^{'}_{i-1}}$ that lead to source string permutation $s^{'}_i$ that are closer to target word order than $s^{'}_{i-1}$; we employ $s^{'}$ language model probability ratios as a measure of word order improvement.

The remainder of the paper is organized as follows. Section 2 provides some background on phrase-based SMT, Section 3 presents the architecture and details of our reordering system, Section 4 reviews related work, Section 5 reports on the experiments done, and Section 6 concludes the article.

2. Baseline system

Current state-of-the-art phrase-based SMT systems [5, 6] start-out from a word-aligned parallel corpus working with (in principle) arbitrarily large phrase pairs (also called blocks) acquired from word-aligned parallel data under a simple definition of translational equivalence [7].

The conditional probabilities of one phrase given its counterpart is estimated as the relative frequency ratio of the phrases in the multiset of phrase-pairs extracted from the parallel corpus and are interpolated log-linearly together with a set of other model estimates:

$$\hat{e}_1^I = \arg\max_{e_1^I} \left\{ \sum_{m=1}^{M} \lambda_m h_m(e_1^I, f_1^J) \right\} \qquad (1)$$

where a feature function h_m refers to a system model, and the corresponding λ_m refers to the relative weight given to this model.

A phrase-based system employs feature functions for a phrase pair translation model, a language model, a reordering model, and a model to score translation hypothesis according to length. The weights λ_m are optimized for system performance [8] as measured by BLEU [9].

Apart from the novel syntax-based reordering model, we consider two reordering methods that are widely used in phrase-based systems: a simple distance-based reordering and a lexicalized block-oriented data-driven reordering model [10].

3. Source reordering system

Given a word-aligned parallel corpus, we define the source string permutation as the task of learning to unfold the crossing alignments between sentence pairs in the parallel corpus. Let be given a source-target sentence pair $s \rightarrow t$ with word alignment set a between their words. Unfolding the crossing instances in a should lead to as monotone an alignment $a^{'}$ as possible between a permutation $s^{'}$ of s and the target string t. Conducting such a "monotonization" on the parallel corpus gives two parallel corpora: (1) a source-to-permutation parallel corpus ($s \rightarrow s^{'}$) and (2) a source permutation-to-target parallel corpus ($s^{'} \rightarrow t$). The latter corpus is word-aligned automatically again and used for training a phrase-based translation system, while the former corpus is used for training our model for pre-translation source permutation via parse tree transductions.

In itself, the problem of permuting the source string to unfold the crossing alignments is computationally intractable (see [4]). However, different kinds of constraints can be made on unfolding the crossing alignments in a. A common approach in hierarchical SMT is to assume that the source string has a binary parse tree, and the set of eligible permutations is defined by binary ITG transductions on this tree. This defines permutations that can be obtained only by at most inverting pairs of children under nodes of the source tree.

3.1. Source Permutation via Syntactic Transfer

Given a parallel corpus with string pairs $s \rightarrow t$ with word alignment a, we create a *source permuted* parallel corpus $s \rightarrow s^{'}$ by unfolding the crossing alignments in a: this is done by scanning the string s from left to right and moving words involved in crossing alignments to positions where the crossing alignments are unfolded). The source strings s are parsed, leading to a single parse tree τ_s per source string.

Our model aims at learning from the source permuted parallel corpus $s \rightarrow s^{'}$ a probabilistic optimization $\arg\max_{\pi(s)} P(\pi(s) \mid s, \tau_s)$. We assume that the set of permutations $\{\pi(s)\}$ is defined through a finite set of local transductions over the tree τ_s. Hence, we view the permutations leading from s to $s^{'}$ as a sequence of local tree transductions $\tau_{s^{'}_0} \rightarrow \ldots \rightarrow \tau_{s^{'}_n}$, where $s^{'}_0 = s$ and $s^{'}_n = s^{'}$, and each transduction $\tau_{s^{'}_{i-1}} \rightarrow \tau_{s^{'}_i}$ is defined using a tree transduction operation that at most permutes the children of a single node in $\tau_{s^{'}_{i-1}}$ as defined next.

A local transduction $\tau_{s^{'}_{i-1}} \rightarrow \tau_{s^{'}_i}$ is modelled by an operation that applies to a single node with address x in $\tau_{s^{'}_{i-1}}$, labeled N_x, and may permute the ordered sequence of children α_x dominated by node x. This constitutes a direct generalization of the ITG binary inversion transduction operation. We assign a conditional probability to each such local transduction:

$$P(\tau_{s^{'}_i} \mid \tau_{s^{'}_{i-1}}) \approx P(\pi(\alpha_x) \mid N_x \rightarrow \alpha_x, C_x) \qquad (2)$$

where $\pi(\alpha_x)$ is a permutation of α_x (the ordered sequence of node labels under x) and C_x is a local tree context of node x in tree $\tau_{s^{'}_{i-1}}$. One wrinkle in this definition is that the number of possible permutations of α_x is factorial in the length of α_x. Fortunately, the source permuted training data exhibits only a fraction of possible permutations even for longer α_x sequences. Furthermore, by conditioning the probability on local context, the general applicability of the permutation is restrained.

Given this definition, we define the probability of the sequence of local tree transductions $\tau_{s^{'}_0} \rightarrow \ldots \rightarrow \tau_{s^{'}_n}$ as

$$P(\tau_{s^{'}_0} \rightarrow \ldots \rightarrow \tau_{s^{'}_n}) = \prod_{i=1}^{n} P(\tau_{s^{'}_i} \mid \tau_{s^{'}_{i-1}}) \qquad (3)$$

The problem of calculating the most likely permutation under this transduction model is made difficult by the fact that different transduction sequences may lead to the same permutation, which demands summing over these sequences. Furthermore, because every local transduction conditions on local context of an intermediate tree, this quickly risks becoming intractable (even when we use packed forests). In practice we take a pragmatic approach and greedily select at every intermediate point $\tau_{s^{'}_{i-1}} \rightarrow \tau_{s^{'}_i}$ the single most likely local transduction that can be conducted on any node of the current intermediate tree $\tau_{s^{'}_{i-1}}$ using an interpolation of the term in Equation 2 with string probability ratios as follows:

$$P(\pi(\alpha_x) \mid N_x \rightarrow \alpha_x, C_x)^{\alpha} \times \left(\frac{P(s'_{i-1})}{P(s'_i)}\right)^{\beta}$$

where α and β are heuristically adjusted binary values.

The rationale behind this log-linear interpolation is that our source permutation approach aims at finding the optimal permutation s' of s that can serve as input for a subsequent translation model. Hence, we aim at tree transductions that are syntactically motivated that also lead to improved string permutation. In this sense, the tree transduction definitions can be seen as an efficient and syntactically informed way to define the space of possible permutations.

We estimate the string probabilities $P(s'_i)$ using 5-gram language models trained on the monotonized corpus s'. We estimate the conditional probability $P(\pi(\alpha_x) \mid N_x \rightarrow \alpha_x, C_x)$ using a Maximum-Entropy framework, where feature functions are defined to capture the permutation as a class, the node label N_x and its head POS tag, the child sequence α_x together with the corresponding sequence of head POS tags and other features corresponding to different contextual information.

We were particularly interested in those linguistic features that motivate reordering phenomena from the syntactic and linguistic perspective. The features that were used for training the permutation system are extracted for every internal node of the source tree that has more than one child:

- *Local tree topology.* Sub-tree instances that include parent node and the ordered sequence of child node labels.
- *Dependency features.* Features that determine the POS tag of the head word of the current node, together with the sequence of POS tags of the head words of its child nodes.
- *Syntactic features.* Three binary features from this class describe: (1) whether the parent node is a child of the node annotated with the same syntactic category, (2) whether the parent node is a descendant of the node annotated with the same syntactic category.

We did not use any language-specific features neither of Chinese, nor for English first of all to provide scalability of the reordering system. In the experiments with EuroParl data (see [3]), we trained separate models for the categories with high number of crossing alignments, along with combinatorial models piling isolated models in certain combinations. One issue we addressed when constructing the maximum entropy reordering system is the adaptation of proposed reordering technique to the task of speech translation. The challenges we were trying to overcome were the specificity of DIALOG corpus (short and ungrammatical sentences), and high sparsity of the trained model. The preliminary results showed that system performance increases if all the feature functions are piled up into a single model.

Example of syntactic reordering for a Chinese-to-English translation can be found in Figure 2. It illustrates the alignment-driven swapping of NP (that spans DNP and NP) and VP (that spans VV and NP) sub-trees on the source (Chinese) side of the parallel corpus. The $NP \leftrightarrow VP$ swapping leads to monotonization of the alignment.

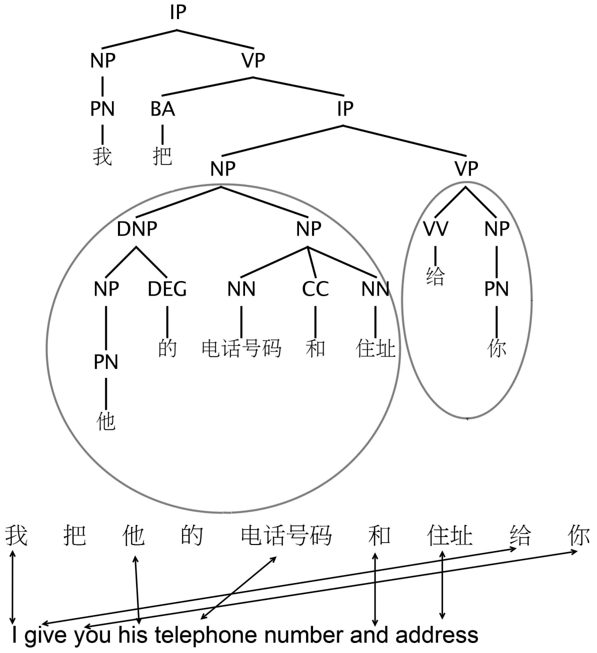
IP
NP
VP
PN
BA
IP
我
把
NP
VP
DNP
NP
VV
NP
NP
DEG
NN
CC
NN
给
PN
PN
的
电话号码
和
住址
你
他
我 把 他 的 电话号码 和 住址 给 你
I give you his telephone number and address

Figure 2: *Example of the reordering algorithm application.*

4. Related work

The integration of linguistic syntax into SMT systems offers a potential solution to reordering problem. For example, syntax is successfully integrated into hierarchical SMT [11]. In [12], a set of tree-string channel operations is defined over the parse tree nodes, while reordering is modeled by permutations of children nodes. Similarly, the tree-to-string syntax-based transduction approach offers a complete translation framework [13].

The idea of augmenting SMT by a reordering step prior to translation has often been shown to improve translation quality. Clause restructuring performed with handcrafted reordering rules for German-to-English and Chinese-to-English tasks are presented in [2] and [14], respectively. In [15, 16] word reordering is addressed by exploiting syntactic representations of source and target texts.

In [1] source and target word order harmonization is done using well-established SMT techniques and without the use of syntactic knowledge. Other reordering models provide the decoder with multiple word orders. For example, the MaxEnt reordering model described in [17] provides a hierarchical phrasal reordering system integrated within a CKY-style decoder. In [18] the authors present an extension of the famous

MSD model [10] able to handle long-distance word-block permutations. Coming up-to-date, in [19] an effective application of data mining techniques to syntax-driven source reordering for MT is presented.

Different syntax-based reordering systems can be found in [20] and [21]. In both systems, reordering rules capable to capture many important word order transformations are automatically learned and applied in the preprocessing step.

Recently, Tromble and Eisner [4] define source permutation as learning source permutations; the model works with a preference matrix for word pairs, expressing preference for their two alternative orders, and a corresponding weight matrix that is fit to the parallel data. The huge space of permutations is then structured using a binary synchronous context-free grammar (Binary ITG) with $O(n^3)$ parsing complexity, and the permutation score is calculated recursively over the tree at every node as the accumulation of the relative differences between the word-pair scores taken from the preference matrix. Application to German-to-English translation exhibits some performance improvement.

Our work is in the general learning direction taken in [4] but differs both in defining the space of permutations, using local probabilistic tree transductions, as well as in the learning objective aiming at scoring permutations based on a log-linear interpolation of a local syntax-based model with a global string-based (language) model.

5. Experiments

In this section we report the experimental work conducted for IWSLT 20010 shared task. ILLC-UvA participated in the Chinese↔English DIALOG tasks.

5.1. Corpus

The experiments with the Chinese to English MT were carried out on the DIALOG Chinese-English data, briefly presented on the corresponding IWSLT 2010 web-page[1]. A detailled statistics of the DIALOG corpus proposed to the participants of the evaluation campaign can be found in Tables 1 and 2. ASL stands for average sentence length.

For Chinese-to-English task, a concatenation of *dev8* and *dev10* was chosen to optimize log-linear weights in the system, and *dev9* was used as an internal test set. For English-to-Chinese system, *dev12* was used to optimize system weights, and *dev11* for internal testing.

5.2. Corpus processing

For internal development work, the Chinese portion of the corpus was provided segmented by words. The preprocessing included deletion of punctuation marks. For the English portion, true case and punctuation marks were removed from all parallel corpora (train, develop, test, and references). For the final evaluation test set, punctuation marks and true case were included by using the SRILM 'disambig' tool following the instruction from the IWSLT 2010 web-page.

Data		Sent.	Words	Voc.	ASL	Refs.
train	Zh	10K	102K	11K	10.16	1
DIALOG	En	10K	118K	6K	11.77	1
train	Zh	20K	172K	8K	8.59	1
BTEC	En	20K	153K	13K	7.66	1
dev1	Zh	506	3.4K	880	6.63	16
dev2	Zh	500	3.5K	920	6.90	16
dev3	Zh	506	3.8K	931	7.44	16
dev4	Zh	489	5.7K	1.1K	11.69	7
dev5	Zh	500	6K	1.3K	12.13	7
dev6	Zh	489	3.1K	881	6.48	6
dev7	Zh	507	3.3K	888	6.61	16
dev8	Zh	246	1.5K	288	6.28	4
dev9	Zh	504	3.0K	392	6.01	7
dev10	Zh	200	2.1K	379	10.70	4
test09.ASR	Zh	405	4.4K	758	10.76	-
test09.CRR	Zh	405	4.6K	653	11.26	-
test10.ASR	Zh	532	4.6K	934	8.60	-
test10.CRR	Zh	532	4.3K	900	8.20	-

Table 1: *Chinese-English corpus.*

Data		Sent.	Words	Voc.	ASL	Refs.
train	En	10K	118K	6K	11.77	1
DIALOG	Zh	10K	102K	11K	10.16	1
train	En	20K	153K	13K	7.66	1
BTEC	Zh	20K	172K	8K	8.59	1
dev3	En	506	3.1K	1.1K	6.16	7
dev10	En	251	1.3K	1.3K	5.14	7
dev11	En	498	2.9K	499	5.82	7
dev12	En	210	2.5K	619	11.77	4
test09.ASR	En	393	4.4K	631	11.28	-
test09.CRR	En	393	4.3K	569	10.99	-
test10.ASR	En	453	5.3K	896	11.70	-
test10.CRR	En	453	5.9K	870	11.05	-

Table 2: *English-Chinese corpus.*

5.3. Training data selection

Apart from the bigger training corpora, there were several small datasets used as development and test corpora in previous campaigns, which were proposed for the participants of the evaluation. Selection of training data was one of the core issues of system construction since these datasets were provided with different numbers of reference translations.

We followed three alternative strategies to accurately select the training material:

[1] http://iwslt2010.fbk.eu/node/33

1. First, we used only the concatenations of BTEC and DIALOG corpora only for training (system "Train only").

2. Second, we extracted individual phrase tables for sets with coinciding number of references (for example, one phrase table was extracted for *dev1*, *dev2*, *dev3*, *dev7* for Chinese-to-English task, one for *dev4* and *dev5*; and one for *dev6*). We then used the Moses capability to use of multiple translation tables during decoding[2]. Two options are possible: (1) translation options are collected from one table, and additional options are collected from the other tables (system "Phrase tables"), and (2) there is an additional table that consists of the intersection of the initial phrase tables, shared phrase pairs are removed from initial tables (system "Phrase tables+intersection"). In the second case, the decoder uses the tables, if the bilingual phrase can be found in them, otherwise it uses only the table where it can find the phrase.

3. Third, we used the target-side language model trained on the concatenation of the DIALOG and BTEC corpora to select a single best reference among the sets of all possible translations according to the highest perplexity. Then, the selected references and its source counterpart is concatenated with the training corpus and used to train the translation model (system "Train+best reference").

The BLEU scores (development and internal test datasets) reflecting different strategies' application can be found in Tables 3 and 4. The systems were built following instructions on the MOSES web-site (`http://www.statmt.org/moses/`) as described in sub-section 5.4.

Strategy	BLEU Dev	BLEU Test
Train only	42.16	33.39
Phrase tables	43.06	35.15
Phrase tables+intersection	42.73	32.35
Train+best reference	42.73	36.07

Table 3: *Different strategies of training data selection. BLEU scores. Chinese-to-English translation.*

5.4. Experimental setup

The SMT system used in the experiments is implemented with standard tools:

- GIZA++/mkcls [8, 22] for word alignment.
- SRI LM [23] for language modeling. A 3-gram target language model was estimated and smoothed with modified Kneser-Ney discounting.
- MOSES [24] to build an unfactored translation system with a MSD reordering model [10] enabled.
- the Stanford parser [25] was used as a source-side parsing engine[3].
- For maximum entropy modeling we used the maxent toolkit[4].

Strategy	BLEU Dev	BLEU Test
Train only	29.15	32.07
Phr.tables all	27.84	31.53
Phr.tables intersection	28.10	31.02
Train+best reference	29.21	32.54

Table 4: *Different strategies of training data selection. BLEU scores. English-to-Chinese translation.*

The discriminative syntactic reordering model is applied to reorder training, development, and test corpora. A Moses-based translation system (corpus realignment included) is then trained using reordered input.

5.5. Experiments and submissions

For each translation task we submitted translations for 3 different systems, which we call "Primary", "Secondary1", and "Secondary2".

The parameters that we adjusted to fit the task were:

- α and β indexes, that adjust the involvement of MaxEnt and language models into tree transduction;
- the order of the idealized source-side language model, navigating the reordering process;
- the data selection strategy.

The ranking of submission was done according to the results shown on the internal testing.

Table 5 shows the configurations of the systems that we experimented with, BLEU scores for the development and test datasets, and our choice for submissions. It is worthwhile to mention that the average number of reorderings was around 1.58 per sentence.

5.6. Official results

In Table 6 we report the BLEU scores obtained by our systems in the official evaluation.

Notice that we provide *"no_case+no_punc"* evaluation specifications only. Along with our systems' scores, we indicate the best system's score and the rank of our systems among all primary runs.

[2] `http://www.statmt.org/moses/?n=Moses.AdvancedFeatures\#ntoc15`

[3] The parser was trained on the English treebank set provided with 14 syntactic categories and 48 POS tags.

[4] `http://homepages.inf.ed.ac.uk/lzhang10/maxent_toolkit.html`

System	α	β	LM order	Data selection	Submission	BLEU Dev	BLEU Test
Chinese-to-English							
1	0	0	-	Train+best reference	Primary	42.73	36.07
2	0	1	-		-	41.80	35.03
3	1	0	3		-	41.84	35.17
4	1	0	4		Secondary2	42.12	35.61
5	1	1	4		Secondary1	41.91	35.61
English-to-Chinese							
1	0	0	-	Train+best reference	Secondary1	29.15	32.17
2	0	1	-		-	28.52	31.84
3	0	1	-		-	28.15	31.93
4	1	1	3		Primary	29.91	32.76
5	1	1	4		Secondary2	28.65	32.15

Table 5: *Summary of experimental results and configurations of submitted systems.*

Task	UvA-ILLC primary	Best	Rank
Chinese-to-English			
IWSLT09.ASR	23.14	34.01	10/11
IWSLT09.CRR	25.80	37.21	10/11
IWSLT10.ASR	14.31	22.20	9/11
IWSLT10.CRR	15.25	24.58	9/11
English-to-Chinese			
IWSLT09.ASR	24.94	38.57	9/11
IWSLT09.CRR	29.26	49.61	9/11
IWSLT10.ASR	17.27	30.80	9/11
IWSLT10.CRR	19.13	37.67	10/11

Table 6: *Official BLEU scores for IWSLT 2010 Chinese-to-English and English-to-Chinese DIALOG tasks.*

6. Conclusions and future work

This paper has presented the ILLC-UvA translation system for Chinese↔English DIALOG tasks proposed to the participants of the IWSLT 2010 evaluation campaign.

The main novelty is that we introduced a tree-based reordering model that aims at monotonizing the word order of source and target languages as a pre-translation step. Our model avoids complete generalization of reordering instances by using tree contexts and limiting the permutations to data instances. From a learning perspective, our work shows that navigating a large space of intermediate tree transformations can be conducted effectively using both the source-side syntactic tree and a language model of the idealized (target-like) source-permuted language.

The secondary distinguishing feature of our system is an accurate training data selection, that shows clear improvements in BLEU score over the baseline configuration.

7. Acknowledgements

Both authors are supported by a VIDI grant (nr. 639.022.604) from The Netherlands Organization for Scientific Research (NWO).

8. References

[1] M. R. Costa-jussà and J. A. R. Fonollosa, "Statistical machine reordering," in *Proceedings of HLT/EMNLP'06*, 2006, pp. 70–76.

[2] M. Collins, P. Koehn, and I. Kučerová, "Clause restructuring for statistical machine translation," in *Proceedings of ACL'05*, 2005, pp. 531–540.

[3] M. Khalilov and K. Sima'an, "A discriminative syntactic model for source permutation via tree transduction," in *Proc. of the Fourth Workshop on Syntax and Structure in Statistical Translation (SSST-4) at COLING'10*, Beijing (China), August 2010, pp. 92–100.

[4] R. Tromble and J. Eisner, "Learning linear ordering problems for better translation," in *Proceedings of EMNLP'09*, 2009, pp. 1007–1016.

[5] F. Och and H. Ney, "Discriminative training and maximum entropy models for statistical machine translation," in *Proceedings of ACL'02*, 2002, pp. 295–302.

[6] P. Koehn, F. Och, and D. Marcu, "Statistical phrase-based machine translation," in *Proceedings of the HLT-NAACL 2003*, 2003, pp. 48–54.

[7] R. Zens, F. Och, and H. Ney, "Phrase-based statistical machine translation," in *Proceedings of KI: Advances in Artificial Intelligence*, 2002, pp. 18–32.

[8] F. Och, "Minimum error rate training in statistical machine translation," in *Proceedings of ACL'03*, 2003, pp. 160–167.

[9] K. Papineni, S. Roukos, T. Ward, and W. Zhu, "Bleu: a method for automatic evaluation of machine translation," in *Proceedings of ACL'02*, 2002, pp. 311–318.

[10] C. Tillman, "A unigram orientation model for statistical machine translation," in *Proceedings of HLT-NAACL'04*, 2004, pp. 101–104.

[11] A. Zollmann and A. Venugopal, "Syntax augmented machine translation via chart parsing," in *Proceedings of NAACL'06*, 2006, pp. 138–141.

[12] K. Yamada and K. Knight, "A syntax-based statistical translation model," in *Proceedings of ACL'01*, 2001, pp. 523–530.

[13] M. Galley, J. Graehl, K. Knight, D. Marcu, S. DeNeefe, W. Wang, and I. Thaye, "Scalable inference and training of context-rich syntactic translation models," in *Proc. of COLING/ACL'06*, 2006, pp. 961–968.

[14] C. Wang, M. Collins, and P. Koehn, "Chinese syntactic reordering for statistical machine translation," in *Proceedings of EMNLP-CoNLL'07*, 2007, pp. 737–745.

[15] F. Xia and M. McCord, "Improving a statistical MT system with automatically learned rewrite patterns," in *Proceedings of COLING'04*, 2004, pp. 508–514.

[16] M. Khalilov, "New statistical and syntactic models for machine translation," Ph.D. dissertation, Universitat Politècnica de Catalunya, October 2009.

[17] D. Xiong, Q. Liu, and S. Lin, "Maximum entropy based phrase reordering model for statistical machine translation," in *Proceedings of ACL'06*, 2006, pp. 521–528.

[18] M. Galley and C. D. Manning, "A simple and effective hierarchical phrase reordering model," in *Proceedings of EMNLP'08*, 2008, pp. 848–856.

[19] A. PVS, "A data mining approach to learn reorder rules for SMT," in *Proceedings of NAACL/HLT'10*, 2010, pp. 52–57.

[20] K. Visweswariah, J. Navratil, J. Sorensen, V. Chenthamarakshan, and N. Kambhatla, "Syntax based reordering with automatically derived rules for improved statistical machine translation," in *Proc. of COLING'10*, Beijing, China, 2010, pp. 1119–1127.

[21] D. Genzel, "Aumotatically learning source-side reordering rules for large scale machine translation," in *Proc. of COLING'10*, Beijing, China, 2010, pp. 376–384.

[22] F. Och, "An efficient method for determining bilingual word classes," in *Proceedings of ACL 1999*, 1999, pp. 71–76.

[23] A. Stolcke, "SRILM: an extensible language modeling toolkit," in *Proceedings of SLP'02*, 2002, pp. 901–904.

[24] P. Koehn, H. Hoang, A. Birch, C. Callison-Burch, M. Federico, N. Bertoldi, B. Cowan, W. Shen, C. Moran, R. Zens, C. Dyer, O. Bojar, A. Constantin, and E. Herbst, "Moses: open-source toolkit for statistical machine translation," in *Proceedings of ACL 2007*, 2007, pp. 177–180.

[25] D. Klein and C. Manning, "Accurate unlexicalized parsing," in *Proceedings of the 41st Annual Meeting of the ACL'03*, 2003, pp. 423–430.