NTT-NAIST Syntax-based SMT Systems for IWSLT 2014

Katsuhito Sudoh[*], *Graham Neubig*[†], *Kevin Duh*[†], *Katsuhiko Hayashi*[*]

[*]NTT Communication Science Laboratories, Seika-cho, Kyoto, Japan
[†]Nara Institute of Science and Technology (NAIST), Ikoma-shi, Nara, Japan
sudoh.katsuhito@lab.ntt.co.jp

Abstract

This paper presents NTT-NAIST SMT systems for English-German and German-English MT tasks of the IWSLT 2014 evaluation campaign. The systems are based on generalized minimum Bayes risk system combination of three SMT systems using the forest-to-string, syntactic pre-ordering, and phrase-based translation formalisms. Individual systems employ training data selection for domain adaptation, truecasing, compound word splitting (for German-English), interpolated n-gram language models, and hypotheses rescoring using recurrent neural network language models.

1. Introduction

Spoken language is a very important and also challenging target for machine translation (MT). MT tasks in the IWSLT evaluation campaign focus on the translation of TED Talks subtitles. These subtitles tend to be clean transcriptions with few disfluencies, and the talks themselves are logically and syntactically well-organized compared to casual conversations.

In order to take advantage of this fact, our system this year use syntax-based statistical machine translation (SMT) techniques, which allow for the use of source-side syntactic knowledge to improve translation accuracy. Specifically, we use forest-to-string (F2S) translation and syntax-based pre-ordering. The overall system was based on a combination of three systems based on F2S, pre-ordering, and standard PBMT, and includes domain adaptation of translation and language models, rescoring using neural network language models, and compound splitting for German.

Specifically comparing to our system from last year's competition [1], we have made two improvements. The first is that we tested a new hypergraph search algorithm [2] in the F2S system, and compare it to the more traditional method of cube pruning. The second is that this year we attempted to extract pre-ordering rules automatically from parallel corpora, as opposed to hand-designing preordering rules based on linguistic intuition.

This paper presents details of our systems and reports the official results together with some detailed discussions on contributions of the techniques involved.

2. Individual Translation Methods

We use three different translation methods and combine the results through system combination. Each of the three methods is described in this section, focusing especially on our new attempts this year on forest-to-string and pre-ordering.

2.1. Forest-to-String Machine Translation

In our previous year's submission to IWSLT, we achieved promising results using the forest-to-string machine translation (F2S; [3]) framework. F2S is a generalization of tree-to-string machine translation (T2S; [4]) that performs translation by first syntactically parsing the source sentence, then translating from sub-structures of a packed forest of potential parses to a string in the target language.

We have previously found that F2S produces highly competitive results for language pairs with large divergence in syntax such as Japanese-English or Japanese-Chinese [5]. However, we have also found that there are several elements that must be appropriately handled to achieve high translation accuracy using syntax-driven methods [6], one of which is search. In the F2S component of our submission to IWSLT this year, we experimented with two different search algorithms to measure the effect that search has on the German-English and English-German pairs.

As the first algorithm, we use the standard method for search in tree-based methods of translation: *cube pruning* [7]. For each edge to be expanded, cube pruning sorts the child hypotheses in descending order of probability, and at every step pops the highest-scoring hypothesis off the stack, calculates its language model scores, and adds the popped, scored edge to the hypergraph. It should be noted that the LM scores are not calculated until after the edge is popped, and thus the order of visiting edges is based on only an LM-free approximation of the true edge score, resulting in search errors.

In our F2S system this year, we test a new method of *hypergraph search* [2], which aims to achieve better search accuracy by considering the characteristics of LM states when deciding the order in which to calculate edges. Particularly, it exploits the fact that states with identical unigram contexts are likely to have similar probabilities, and groups these together at the beginning of the search. It then proceeds to split these states into bi-gram or higher order contexts gradu-

Proceedings of the 11th International Workshop on Spoken Language Translation
Lake Tahoe, December 4th and 5th, 2014

ally, refining the probability estimates until the limit on number of stack pops is reached. In our previous work [6] we have found that hypergraph search achieved superior results to cube pruning, and we hypothesize that these results will carry over to German-English and English-German as well.

2.2. Syntax-based Pre-ordering

Pre-ordering is a method that attempts to first reorder the source sentence into a word order that is closer to the target, then translate using a standard method such as PBMT. We used hand-crafted German-English pre-ordering rules [8] in our submission last year. This year's system uses an automatic method to extract domain-dependent pre-ordering rules, avoiding the time-consuming effort required for creating hand-crafted rules. The pre-ordering method is basically similar to [9], but is limited to reordering of child nodes in syntactic parse trees rather than rewriting and word insertion.

Since the pre-ordering does not work perfectly in all cases, we allow for further reordering in the PBMT system that translates the preordered sentences. The reordering limit of this system is chosen experimentally using held-out data (dev. set BLEU in this paper).

2.2.1. Reordering Pattern Extraction

A reordering pattern represents a reordering of child nodes in a source language parse tree, determined by word alignment. The reordering pattern is similar to a tree-based translation pattern called *frontier graph fragments*, which form the most basic unit in tree-based translation [10], but only holds reordering information on the non-terminal child nodes. A reordering pattern can be extracted from an *admissible* node [11] in the parse tree that covers a distinct contiguous spans in the corresponding target language sentences. Since such a reordering pattern only is constrained by the syntactic labels on the parent and child nodes, we consider several attributes of reordering patterns: syntactic labels of its grand-parent, left and right siblings of the parent, and surface forms of its child nodes (only when the child is a part-of-speech node).

2.2.2. Deterministic Pre-ordering

In order to make the pre-ordering deterministic, we use reordering rules from dominant reordering patterns that agree with more than 75% on the same source language subtrees. Here, additional attributes define more specific rules that are not applied to the subtrees with different attributes.

We apply these reordering rules greedily to the syntactic parse tree in descending order of preference from specific (more attributes) to general (less attributes) rules. If different rules with the same number of attributes can be applied, the most probable one is chosen. More details about the method can be found in [9].

2.3. Standard Phrase-based Translation

Phrase-based machine translation (PBMT; [12]) models the translation process by splitting the source sentence into phrases, translating the phrases into target phrases, and reordering the phrases into the target language order. PBMT is currently the most widely used method in SMT as it is robust, does not require the availability of linguistic analysis tools, and achieves high accuracy, particularly for languages with similar syntactic structure.

3. Additional System Enhancements

Here we review techniques that were used in our submission last year [1] and also describe some of our new attempts that were not effective in our pilot test and not included in the final system.

3.1. Training Data Selection

The target TED domain is different in both style and vocabulary from many of the other bitexts, e.g. Europarl, CommonCrawl (which we collectively call "general-domain" data). To address this domain adaption problem, we performed adaptation training data selection using the method of [13].[1] The intuition is to select general-domain sentences that are similar to in-domain text, while being dis-similar to the average general-domain text.

To do so, one defines the score of an general-domain sentence pair (e, f) as [14]:

$$[IN_E(e) - GEN_E(e)] + [IN_F(f) - GEN_F(f)] \quad (1)$$

where $IN_E(e)$ is the *length-normalized* cross-entropy of e on the English in-domain LM. $GEN_E(e)$ is the length-normalized cross-entropy of e on the English general-domain LM, which is built from a sub-sample of the general-domain text. Similarly, $IN_F(f)$ and $GEN_F(f)$ are the cross-entropies of f on Foreign-side LM. Finally, sentence pairs are ranked according to Eq. 1 and those with scores lower than some empirically-chosen threshold are added together with the in-domain bitext for translation model training. Here, the LMs are Recurrent Neural Network Language Models (RNNLMs), which have been shown to outperform n-gram LMs in this problem [13].

3.2. German Compound Word Splitting

German compound words present sparsity challenges for machine translation. To address this, we split German words following the general approach of [15]. The idea is to split a word if the geometric average of its subword frequencies is larger than whole word frequency. In our implementation, for each word, we searched for all possible decompositions into two sub-words, considering the possibility of deleting common German fillers "e", "es", and "s" (as in "Arbeit+s+tier"). The unigram frequencies for the subwords and

[1] Code/scripts available at http://cl.naist.jp/∼kevinduh/a/acl2013

whole word is computed from the German part of the bitext. This simple algorithm is especially useful for handling out-of-vocabulary and rare compound words that have high frequency sub-words in the training data. For the F2S system, sub-words are given the same POS tag as the original whole word.

In the evaluation campaign, we performed compound splitting only in the German-to-English task. We do not attempt to split German words for the English-to-German task, since it is non-trivial to handle recombination of German split words after reordering and translation.

3.3. RNNLM Rescoring

Continuous-space language models using neural networks have attracted recent attention as a method to improve the fluency of output of MT or speech recognition. In our system, we used the recurrent neural network language model (RNNLM) of [16].[2] This model uses a continuous space representation over the language model state that is remembered throughout the entire sentence, and thus has the potential to ensure the global coherence of the sentence to the greater extent than simpler n-gram language models.

We incorporate the RNNLM probabilities through rescoring. For each system, we first output a 10,000-best list, then calculate the RNNLM log probabilities and add them as an additional feature to each translation hypothesis. We then re-run a single MERT optimization to find ideal weights for this new feature, and then extract the 1-best result from the 10,000-best list for the test set according to these new weights. The parameters for RNNLM training are tuned on the dev set to maximize perplexity, resulting in 300 nodes in the hidden layer, 300 classes, and 4 steps of back-propagation through time.

3.4. GMBR System Combination

We used a system combination method based on Generalized Minimum Bayes Risk optimization [17], which has been successfully applied to different types of SMT systems for patent translation [18]. Note that our system combination only picks one hypothesis from an N-best list and does not generate a new hypothesis by mixing partial hypotheses among the N-best.

3.4.1. Theory

Minimum Bayes Risk (MBR) is a decision rule to choose hypotheses that minimize the expected loss. In the task of SMT from a French sentence (f) to an English sentence (e), the MBR decision rule on $\delta(f) \rightarrow e'$ with the loss function L over the possible space of sentence pairs ($p(e, f)$) is denoted as:

$$\underset{\delta(f)}{\operatorname{argmin}} \sum_{e} L(\delta(f)|e)p(e|f) \tag{2}$$

[2] http://www.fit.vutbr.cz/~imikolov/rnnlm/

In practice, we approximate this using N-best list $N(f)$ for the input f.

$$\underset{e' \in N(f)}{\operatorname{argmin}} \sum_{e \in N(f)} L(e'|e)p(e|f) \tag{3}$$

Although MBR works effectively for re-ranking single system hypotheses, it is challenging for system combination because the estimated $p(e|f)$ from different systems cannot be reliably compared. One practical solution is to use uniform $p(e|f)$ but this does not achieve Bayes Risk. GMBR corrects by parameterizing the loss function as a linear combination of sub-components using parameter $\boldsymbol{\theta}$:

$$L(e'|e;\boldsymbol{\theta}) = \sum_{k=1}^{K} \theta_k L_k(e'|e) \tag{4}$$

For example, suppose the desired loss function is "1.0−BLEU". Then the sub-components could be "1.0−precision(n-gram) $(1 \leq n \leq 4)$" and "brevity penalty".

Assuming uniform $p(e|f)$, the MBR decision rule can be denoted as:

$$\begin{aligned} &\underset{e' \in N(f)}{\operatorname{argmin}} \sum_{e \in N(f)} L(e'|e;\boldsymbol{\theta}) \frac{1}{|N(f)|} \\ &= \underset{e' \in N(f)}{\operatorname{argmin}} \sum_{e \in N(f)} \sum_{k=1}^{K} \theta_k L_k(e'|e) \end{aligned} \tag{5}$$

To ensure that the uniform hypotheses space gives the same decision as the original loss in the true space $p(e|f)$, we use a small development set to tune the parameter $\boldsymbol{\theta}$ as follows. For any two hypotheses e_1, e_2, and a reference translation e_r (possibly not in $N(f)$) we first compute the true loss: $L(e_1|e_r)$ and $L(e_2|e_r)$. If $L(e_1|e_r) < L(e_2|e_r)$, then we would want $\boldsymbol{\theta}$ such that:

$$\sum_{e \in N(f)} \sum_{k=1}^{K} \theta_k L_k(e_1|e) < \sum_{e \in N(f)} \sum_{k=1}^{K} \theta_k L_k(e_2|e) \tag{6}$$

so that GMBR would select the hypothesis achieving lower loss. Conversely if e_2 is a better hypothesis, then we want opposite relation:

$$\sum_{e \in N(f)} \sum_{k=1}^{K} \theta_k L_k(e_1|e) > \sum_{e \in N(f)} \sum_{k=1}^{K} \theta_k L_k(e_2|e) \tag{7}$$

Thus, we directly compute the true loss using a development set and ensure that our GMBR decision rule minimizes this loss.

3.4.2. Implementation

We implement GMBR for SMT system combination as follows.

First we run SMT decoders to obtain N-best lists for all sentences in the development set, and extract all pairs of hypotheses where a difference exists in the true loss. Then we optimize θ in a formulation similar to a Ranking SVM [19]. The pair-wise nature of Eqs. 6 and 7 makes the problem amendable to solutions in "learning to rank" literature [20]. We used BLEU as the objective function and the subcomponents of BLEU as features (system identity feature was not used). There is one regularization hyperparameter for the Ranking SVM, which we set by cross-validation over the development set (`dev2010`).

3.5. What Didn't Work Immediately

This year we tried to include a state-of-the-art Neural Network Joint Model (NNJM) [21] to improve the accuracy of translation probability estimation. The model is used to predict a target language word using its three preceding target language words and eleven source language words surrounding its affiliation (the non-NULL source language word aligned to the target language word to be predicted). We used top 16,000 source and target vocabularies in the model and mapped the other words into a single OOV symbol, while the original paper[21] used part-of-speech classes. Although the original paper presented a method for integrating the model with decoding, we used the NNJM for reranking n-best hypotheses in a similar manner as the RNNLM described above. The NNJM gave some improvements from the baseline 1-best in our pilot test, but they were much smaller than those resulting from RNNLM, and when the NNJM was combined with RNNLM we saw no significant gains. One possible reason is the small training data size; the model is very sparse and needs large training data because of its large contexts of fourteen (eleven source and three target) words. The affiliation is very important to predict the target word correctly but it was determined by automatic word alignment (such as GIZA++) and may not always be good enough in our experiments.

We also tried *post-ordering* [22] by shift-reduce reordering [23] for German-to-English. It was not effective in our pilot test even in the first-pass lexical translation, probably due to less effective English-to-German pre-ordering rules.

4. Experiments

We conducted experiments on the English-German and German-English MT tasks using the SMT systems described above developed using the supplied datasets.

4.1. Setup

4.1.1. System Overview

We used three individual SMT systems presented in Section 2: forest-to-string (F2S), phrase-based with pre-ordering (Preorder), and phrase-based without pre-ordering (PBMT). F2S was implemented with Travatar[3] [24] and the phrase-based MT systems were implemented with Moses [25].

For the Travatar rule tables, we used a modified version of Egret[4] as a syntactic parser, and created forests using dynamic pruning including all edges that occurred in the 100-best hypotheses. We trained the parsing model using the Berkeley parser over the Wall Street Journal section of the Penn Treebank[5] for English, and TIGER corpus [26] for German. For model training, the default settings for Travatar were used, with the exception of changing the number of composed rules to 6 with Kneser-Ney smoothing. For search in the F2S models, we used the previously described hypergraph search method.

For the Moses phrase tables, we used standard training settings with Kneser-Ney smoothing of phrase translation probabilities [27].

4.1.2. Translation Models

We trained the translation models using WIT[3] training data (178,526 sentences) and 1,000,000 sentences selected over other bitexts (Europarl, News Commentary, and Common Crawl) by the method described in Section 3.1.

4.1.3. Language Models

We used word 5-gram language models of German and English that were linearly interpolated from several word 5-gram language models trained on different data sources (WIT[3], Europarl, News Commentary, and Common Crawl). The interpolation weights were optimized to minimize perplexity on the development set, using `interpolate-lm.perl` in Moses. Individual language models were trained by SRILM with modified Kneser-Ney smoothing.

4.1.4. Truecaser

In order to maintain the casing of words across languages, we opted to use truecasing (based on the Moses truecaser) on both the source and target sides. Truecasing keeps the case of all words that are not sentence initial, and chooses the case of the sentence initial word based on the most frequent appearance among different cases in the training data.

4.2. Full System Results

Our full system was a GMBR-based combination of F2S, Preorder, and PBMT. Tables 1 and 2 show the official evaluation results for English-to-German and German-to-English tasks, respectively. Among the individual systems, F2S showed the best BLEU and TER, and Preorder was the worst. The poor performance of Preorder was not consistent with our development results on older test sets (discussed later)

[3] `http://www.phontron.com/travatar/`
[4] `https://github.com/neubig/egret/`
[5] `http://www.cis.upenn.edu/~treebank/`

Table 1: Official results for English-to-German (case sensitive). Δ*BestSingle* represents the differences from the results by the best single system (F2S).

System	tst2013		tst2014	
(En-De)	BLEU	TER	BLEU	TER
Combination	.2580	.5386	.2209	.5760
Δ*BestSingle*	+.0097	-.0103	-.0021	-.0100
F2S	.2483	.5489	.2230	.5860
Preorder	.2443	.5567	.2112	.5947
PBMT	.2453	.5528	.2150	.5906

Table 2: Official results for German-to-English (case sensitive).

System	tst2013		tst2014	
(De-En)	BLEU	TER	BLEU	TER
Combination	.2781	.5162	.2377	.5643
Δ*BestSingle*	+.0070	-.0224	+.0030	-.0180
F2S	.2711	.5386	.2347	.5823
Preorder	.2646	.5425	.2208	.5914
PBMT	.2671	.5422	.2229	.5885

Table 3: Percentages of individual system outputs chosen by system combination.

System	En-De		De-En	
	tst2013	tst2014	tst2013	tst2014
F2S	16.11	19.16	57.59	54.24
Preorder	49.14	50.34	39.69	42.72
PBMT	34.74	30.50	1.39	3.04

and our last year's results with hand-crafted rules [1]. The GMBR combination further improved BLEU and TER compared to those of F2S, except for BLEU in tst2014. The improvement in TER was large, about 1% in English-to-German and 2% in German-to-English, compared to an at most 1% gain in BLEU.

Table 3 shows the contributions of individual systems in the system combination, by percentages of *chosen* system outputs. As we discussed in our system description paper last year [1], the GMBR system combination works as voting over n-best hypotheses from different systems. The results in Table 3 indicate the best F2S system contributed little in English-German and the worst Preorder system contributed about a half of the system combination outputs. There were large difference between these results and our last year's results, but we do not yet have a solid answer for the reason. One possibility is the inconsistency between the training condition (Preorder worked well) and the test condition (Preorder worked poorly) as discussed later in detail.

Table 4: Results on old IWSLT test sets for English-to-German (case sensitive). Scores in **bold** indicate the best individual system results.

System	tst2010		tst2011		tst2012	
(En-De)	BLEU	TER	BLEU	TER	BLEU	TER
Combi.	.2516	.6309	.2714	.5870	.2388	.6380
F2S	**.2487**	**.6452**	**.2670**	**.5989**	**.2306**	**.6545**
Preorder	.2412	.6523	.2639	.6043	.2274	.6601
PBMT	.2419	.6509	.2634	.6031	.2280	.6575

Table 5: Results on old IWSLT test sets for German-to-English (case sensitive). Scores in **bold** indicate the best individual system results.

System	tst2010		tst2011		tst2012	
(De-En)	BLEU	TER	BLEU	TER	BLEU	TER
Combi.	.3155	.5583	.3711	.4949	.3144	.5515
F2S	.3037	.5901	.3465	.5313	.3028	.5812
Preorder	**.3065**	**.5730**	**.3604**	**.5088**	**.3055**	**.5647**
PBMT	.3043	.5754	.3571	.5119	.3038	.5678

4.3. Detailed Results and Discussions

4.3.1. Evaluation on Old Test Sets

Tables 4 and 5 shows the results on old IWSLT test sets (tst2010, tst2011, tst2012). The results tend to show a different trend than those for tst2013 and tst2014; Specifically looking at the German-to-English task, F2S was the worst and Preorder worked the best on these older data sets, as shown in Table 5.

One possible reason for this difference is the difference in the *original* languages in the older and newer test sets. The official test sets this year (tst2013, tst2014) came from TEDX talks in German, and thus the source German sentences were transcriptions. In contrast, the older test sets (tst2010, tst2011, tst2012) came from TED talks in English, and thus the source German sentences were translations from English. It has been widely noted that translations differ significantly from original texts stylistically (e.g. [28]), and the difference may cause some inconsistencies in syntactic parsing and syntax-based translation. Preorder used only dominant reordering patterns in German extracted from translated German sentences, which were consistent with the TED test sets but not with the TEDX test sets.

4.3.2. Effect of Search on F2S Translation

As mentioned in Section 2.1, we tested two algorithms for search in F2S models, cube pruning, and hypergraph search. In Figure 1 we show the speed and accuracy for both algorithms at various beam sizes for English-German and German-English translation. All results are reported on tst2010, but similar results were found for other sets.

From these results, we can see that given an identical de-

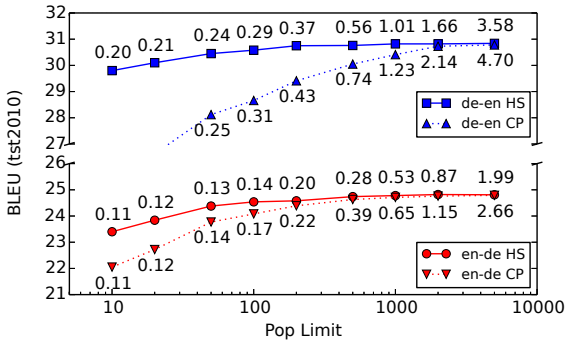

Figure 1: Hypergraph search (HS) and cube pruning (CP) results for F2S translation. Numbers above and below the lines indicate time in seconds/sentence for HS and CP respectively.

coding time, hypergraph search outperforms cube pruning on both language pairs at all beam sizes, especially for smaller beams. This effect was particularly notable for German-English translation. Even when the beam is reduced from 5000 (which was used in our actual submission) to 10, we only see a drop in one BLEU point, but reduce the time required for decoding to 200ms, much of which can be attributed to processing other than search such as rule lookup or file input/output. This is in contrast to cube pruning, which sees a 5.5 BLEU point drop at the same beam size.

5. Conclusion

In this paper, we presented our English-to-German and German-to-English SMT systems using combination of forest-based, pre-ordering, and standard phrase-based MT systems. The forest-based system employed the hypergraph search for efficient translation, and the pre-ordering used automatically-induced rules from the bilingual corpus. The individual systems used training data selection, compound word splitting for German, and RNNLM rescoring, same as our last year's systems. Our results show the forest-to-string SMT was consistently the most effective of the three and can be further improved by GMBR system combination with the results from the other two systems. The pre-ordering was not effective in the 2013 and 2014 test sets in contrast to the older ones.

6. Acknowledgements

We would like to thank the anonymous reviewer for the comments to improve this system description paper. We also appreciate Dr. Jun Suzuki for his support for the use of NNJM in our experiments.

7. References

[1] K. Sudoh, G. Neubig, K. Duh, and H. Tsukada, "NTT-NAIST SMT Systems for IWSLT 2013," in *Proceedings of the 10th International Workshop on Spoken Language Translation (IWSLT 2013)*, Heidelberg, Germany, December 2013.

[2] K. Heafield, P. Koehn, and A. Lavie, "Grouping language model boundary words to speed k–best extraction from hypergraphs," in *Proceedings of the 2013 Annual Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 2013, pp. 958–968.

[3] H. Mi, L. Huang, and Q. Liu, "Forest-based translation," in *Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics (ACL)*, 2008, pp. 192–199.

[4] Y. Liu, Q. Liu, and S. Lin, "Tree-to-string alignment template for statistical machine translation," in *Proceedings of the 44th Annual Meeting of the Association for Computational Linguistics (ACL)*, 2006.

[5] G. Neubig, "Forest-to-string SMT for asian language translation: NAIST at WAT2014," in *Proceedings of the 1st Workshop on Asian Translation (WMT)*, 2014.

[6] G. Neubig and K. Duh, "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 (ACL)*, 2014.

[7] D. Chiang, "Hierarchical phrase-based translation," *Computational Linguistics*, vol. 33, no. 2, pp. 201–228, 2007.

[8] M. Collins, P. Koehn, and I. Kucerova, "Clause Restructuring for Statistical Machine Translation," in *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL'05)*, Ann Arbor, Michigan, June 2005, pp. 531–540.

[9] F. Xia and M. McCord, "Improving a Statistical MT System with Automatically Learned Rewrite Patterns," in *Proceedings of Coling 2004*, Geneva, Switzerland, August 2004, pp. 508–514.

[10] M. Galley, M. Hopkins, K. Knight, and D. Marcu, "What's in a translation rule?" in *HLT-NAACL 2004: Main Proceedings*, Boston, Massachusetts, USA, May 2004, pp. 273–280.

[11] W. Wang, K. Knight, and D. Marcu, "Binarizing Syntax Trees to Improve Syntax-Based Machine Translation Accuracy," in *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, Prague, Czech Republic, June 2007, pp. 746–754.

[12] P. Koehn, F. J. Och, and D. Marcu, "Statistical Phrase-Based Translation," in *Proceedings of the 2003 Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics*, Edmonton, Canada, May 2003, pp. 48–54.

[13] K. Duh, G. Neubig, K. Sudoh, and H. Tsukada, "Adaptation Data Selection using Neural Language Models: Experiments in Machine Translation," in *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, Sofia, Bulgaria, August 2013, pp. 678–683.

[14] A. Axelrod, X. He, and J. Gao, "Domain Adaptation via Pseudo In-Domain Data Selection," in *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*, Edinburgh, Scotland, UK., July 2011, pp. 355–362.

[15] P. Koehn and K. Knight, "Empirical Methods for Compound Splitting," in *Proceedings of the 10th Conference of the European Chapter of the Association for Computational Linguistics*, 2003, pp. 187–194.

[16] T. Mikolov, M. Karafiát, L. Burget, J. Cernockỳ, and S. Khudanpur, "Recurrent neural network based language model," in *Proceedings of the 11th Annual Conference of the International Speech Communication Association (INTERSPEECH 2010)*, Chiba, Japan, 2010, pp. 1045–1048.

[17] K. Duh, K. Sudoh, X. Wu, H. Tsukada, and M. Nagata, "Generalized Minimum Bayes Risk System Combination," in *Proceedings of 5th International Joint Conference on Natural Language Processing*, Chiang Mai, Thailand, November 2011, pp. 1356–1360.

[18] K. Sudoh, K. Duh, H. Tsukada, M. Nagata, X. Wu, T. Matsuzaki, and J. Tsujii, "NTT-UT Statistical Machine Translation in NTCIR-9 PatentMT," in *Proceedings of the 9th NTCIR Conference*, Tokyo, Japan, December 2011.

[19] T. Joachims, "Training Linear SVMs in Linear Time," in *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 2006, pp. 217–226.

[20] C. He, C. Wang, Y.-X. Zhong, and R.-F. Li, ""a survey on learning to rank"," in *Proceedings on 2008 International Conference on Machine Learning and Cybernetics*, vol. 3, 2008, pp. 1734–1739.

[21] J. Devlin, R. Zbib, Z. Huang, T. Lamar, R. Schwartz, and J. Makhoul, "Fast and Robust Neural Network Joint Models for Statistical Machine Translation," in *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Baltimore, Maryland: Association for Computational Linguistics, June 2014, pp. 1370–1380. [Online]. Available: http://www.aclweb.org/anthology/P14-1129

[22] K. Sudoh, X. Wu, K. Duh, H. Tsukada, and M. Nagata, "Post-ordering in statistical machine translation," in *Proceedings of the 13th Machine Translation Summit (MT Summit XIII)*, Xiamen, China, September 2011, pp. 316–323.

[23] K. Hayashi, K. Sudoh, H. Tsukada, J. Suzuki, and M. Nagata, "Shift-Reduce Word Reordering for Machine Translation," in *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, Seattle, Washington, USA, October 2013, pp. 1382–1386.

[24] G. Neubig, "Travatar: A Forest-to-String Machine Translation Engine based on Tree Transducers," in *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, Sofia, Bulgaria, August 2013, pp. 91–96.

[25] 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 the 45th Annual Meeting of the Association for Computational Linguistics Companion Volume Proceedings of the Demo and Poster Sessions*, Prague, Czech Republic, June 2007, pp. 177–180.

[26] S. Brants, S. Dipper, P. Eisenberg, S. Hansen-Schirra, E. h. König, W. Lezius, C. Rohrer, G. Smith, and H. Uszkoreit, "TIGER: Linguistic interpretation of a German corpus," *Research on Language and Computation*, vol. 2, no. 4, pp. 597–620, 2004.

[27] G. Foster, R. Kuhn, and H. Johnson, "Phrasetable Smoothing for Statistical Machine Translation," in *Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing*, Sydney, Australia, July 2006, pp. 53–61.

[28] M. Koppel and N. Ordan, "Translationese and its dialects," in *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics (ACL)*, 2011, pp. 1318–1326.