Edinburgh's Syntax-Based Machine Translation Systems

Maria Nadejde, Philip Williams, and Philipp Koehn
School of Informatics, University of Edinburgh, Scotland, United Kingdom
maria.nadejde@gmail.com, P.J.Williams-2@sms.ed.ac.uk, pkoehn@inf.ed.ac.uk

Abstract

We present the syntax-based string-to-tree statistical machine translation systems built for the WMT 2013 shared translation task. Systems were developed for four language pairs. We report on adapting parameters, targeted reduction of the tuning set, and post-evaluation experiments on rule binarization and preventing dropping of verbs.

1 Overview

Syntax-based machine translation models hold the promise to overcome some of the fundamental problems of the currently dominating phrase-based approach, most importantly handling reordering for syntactically divergent language pairs and grammatical coherence of the output.

We are especially interested in string-to-tree models that focus syntactic annotation on the target side, especially for morphologically rich target languages (Williams and Koehn, 2011).

We have trained syntax-based systems for the language pairs

- English-German,
- German-English,
- Czech-English, and
- Russian-English.

We have also tried building systems for French-English and Spanish-English but the data size proved to be problematic given the time constraints. We give a brief description of the syntax-based model and its implementation within the Moses system. Some of the available features are described as well as some of the pre-processing steps. Several experiments are described and final results are presented for each language pair.

2 System Description

The syntax-based system used in all experiments is the Moses string-to-tree toolkit implementing GHKM rule extraction and Scope-3 parsing previously described in by Williams and Koehn (2012)

2.1 Grammar

Our translation grammar is a synchronous context-free grammar (SCFG) with phrase-structure labels on the target side and the generic non-terminal label X on the source side. In this paper, we write these rules in the form

$$\text{LHS} \rightarrow \text{RHS}_s \mid \text{RHS}_t$$

where LHS is a target-side non-terminal label and RHS_s and RHS_t are strings of terminals and non-terminals for the source and target sides, respectively. We use subscripted indices to indicate the correspondences between source and target non-terminals.

For example, a translation rule to translate the German *Haus* into the English *house* is

$$\text{NN} \rightarrow \textit{Haus} \mid \textit{house}$$

If our grammar also contains the translation rule

$$\text{S} \rightarrow \textit{das ist ein } \text{X}_1 \mid \textit{this is a } \text{NN}_1$$

then we can apply the two rules to an input *das ist ein Haus* to produce the output *this is a house*.

2.2 Rule Extraction

The GHKM rule extractor (Galley et al., 2004, 2006) learns translation rules from a word-aligned parallel corpora for which the target sentences are syntactically annotated. Given a string-tree pair, the set of minimally-sized translation rules is extracted that can explain the example and is consistent with the alignment. The resulting rules can be composed in a non-overlapping fashion in order to cover the string-tree pair.

Two or more minimal rules that are in a parent-child relationship can be composed together to obtain larger rules with more syntactic context. To avoid generating an exponential number of composed rules, several limitation have to be imposed.

One such limitation is on the size of the composed rules, which is defined as the number of non-part-of-speech, non-leaf constituent labels in the target tree (DeNeefe et al., 2007). The corresponding parameter in the Moses implementation is *MaxRuleSize* and its default value is 3.

Proceedings of the Eighth Workshop on Statistical Machine Translation, pages 170–176,
Sofia, Bulgaria, August 8-9, 2013

Another limitation is on the depth of the rules' target subtree. The rule depth is computed as the maximum distance from its root node to any of its children, not counting pre-terminal nodes (parameter *MaxRuleDepth*, default 3).

The third limitation considered is the number of nodes in the composed rule, not counting target words (parameter *MaxNodes*, default 15).

These parameters are language-dependent and should be set to values that best represent the characteristics of the target trees on which the rule extractor is trained on. Therefore the style of the treebanks used for training the syntactic parsers will also influence these numbers. The default values have been set based on experiments on the English-German language pair (Williams and Koehn, 2012). It is worth noting that the German parse trees (Skut et al., 1997) tend to be broader and shallower than those for English. In Section 3 we present some experiments where we choose different settings of these parameters for the German-English language pair. We use those settings for all language pairs where the target language is English.

2.3 Tree Restructuring

The coverage of the extracted grammar depends partly on the structure of the target trees. If the target trees have flat constructions such as long noun phrases with many sibling nodes, the rules extracted will not generalize well to unseen data since there will be many constraints given by the types of different sibling nodes.

In order to improve the grammar coverage to generalize over such cases, the target tree can be restructured. One restructuring strategy is tree binarization. Wang et al. (2010) give an extensive overview of different tree binarization strategies applied for the Chinese-English language pair. Moses currently supports *left binarization* and *right binarization*.

By *left binarization* all the left-most children of a parent node *n* except the right most child are grouped under a new node. This node is inserted as the left child of *n* and receives the label $\bar{n}$. *Left binarization* is then applied recursively on all newly inserted nodes until the leaves are reached. *Right binarization* implies a similar procedure but in this case the right-most children of the parent node are grouped together except the left most child.

Another binarization strategy that is not currently integrated in Moses, but is worth investigating for different language pairs, is *parallel head binarization*.

The result of *parallel binarization* of a parse tree is a binarization forest. To generate a binarization forest node, both right binarization and left binarization are applied recursively to a parent node with more than two children. *Parallel head binarization* is a case of *parallel binarization* with the additional constraint that the head constituent is part of all the new nodes inserted by either left or right binarization steps.

In Section 3 we give example of some initial experiments carried out for the German-English language pair.

2.4 Pruning The Grammar

Decoding for syntax-based model relies on a bottom-up chart parsing algorithm. Therefore decoding efficiency is influenced by the following combinatorial problem: given an input sentence of length *n* and a context-free grammar rule with *s* consecutive non-terminals, there are $\binom{n+1}{s}$ ways to choose subspans, or *application contexts* (Hopkins and Langmead, 2010), that the rule can applied to. The asymptotic running time of chart parsing is linear in this number $O(n^s)$.

Hopkins and Langmead (2010) maintain cubic decoding time by pruning the grammar to remove rules for which the number of potential application contexts is too large. Their key observation is that a rule can have any number of non-terminals and terminals as long as the number of consecutive non-terminal pairs is bounded. Terminals act to anchor the rule, restricting the number of potential application contexts. An example is the rule $X \rightarrow WyYZz$ for which there are at most $O(n^2)$ application contexts, given that the terminals will have a fixed position and will play the role of anchors in the sentence for the non-terminal spans. The number of consecutive non-terminal pairs plus the number of non-terminals at the edge of a rule is referred to as the *scope* of the rule. The scope of a grammar is the maximum scope of any of its rules. Moses implements *scope-3 pruning* and therefore the resulting grammar can be parsed in cubic time.

2.5 Feature Functions

Our feature functions are unchanged from last year. They include the n-gram language model probability of the derivation's target yield, its word

count, and various scores for the synchronous derivation. Our grammar rules are scored according to the following functions:

- $p(\mathrm{RHS}_s|\mathrm{RHS}_t, \mathrm{LHS})$, the noisy-channel translation probability.
- $p(\mathrm{LHS}, \mathrm{RHS}_t|\mathrm{RHS}_s)$, the direct translation probability.
- $p_{lex}(\mathrm{RHS}_t|\mathrm{RHS}_s)$ and $p_{lex}(\mathrm{RHS}_s|\mathrm{RHS}_t)$, the direct and indirect lexical weights (Koehn et al., 2003).
- $p_{pcfg}(\mathrm{FRAG}_t)$, the monolingual PCFG probability of the tree fragment from which the rule was extracted. This is defined as $\prod_{i=1}^{n} p(r_i)$, where $r_1 \ldots r_n$ are the constituent CFG rules of the fragment. The PCFG parameters are estimated from the parse of the target-side training data. All lexical CFG rules are given the probability 1. This is similar to the p_{cfg} feature proposed by Marcu et al. (2006) and is intended to encourage the production of syntactically well-formed derivations.
- $exp(-1/count(r))$, a rule rareness penalty.
- $exp(1)$, a rule penalty. The main grammar and glue grammars have distinct penalty features.

3 Experiments

This section describes details for the syntax-based systems submitted by the University of Edinburgh. Additional post-evaluation experiments were carried out for the German-English language pair.

3.1 Data

We made use of all available data for each language pair except for the Russian-English where the *Commoncrawl* corpus was not used. Table 1 shows the size of the parallel corpus used for each language pair. The English side of the parallel corpus was parsed using the Berkeley parser (Petrov et al., 2006) and the German side of the parallel corpus was parsed using the BitPar parser (Schmid, 2004). For German-English, German compounds were split using the script provided with Moses. The parallel corpus was word-aligned using MGIZA++ (Gao and Vogel, 2008).

All available monolingual data was used for training the language models for each language pair. 5-gram language models were trained using SRILM toolkit (Stolcke, 2002) with modified Kneser-Ney smoothing (Chen and Goodman, 1998) and then interpolated using weights tuned on the newstest2011 development set.

Lang. pair	Sentences	Grammar Size
en-de	4,411,792	31,568,480
de-en	4,434,060	55,310,162
cs-en	14,425,564	209,841,388
ru-en	1,140,359	7,946,502

Table 1: Corpus statistics for parallel data.

The feature weights for each system were tuned on development sets using the Moses implementation of minimum error rate training (Och, 2003). The size of the tuning data varied for different languages depending on the amount of available data. In the case of the the German-English pair a filtering criteria based on sentence level BLEU score was applied which is briefly described in Section 3.5. Table 2 shows the size of the tuning set for each language pair.

Lang. pair	Sentences
en-de	7,065
de-en	2,400
cs-en	10,068
ru-en	1,501

Table 2: Corpus statistics for tuning data.

3.2 Pre-processing

Some attention was given to pre-processing of the English side of the corpus prior to parsing. This was done to avoid propagating parser errors to the rule-extraction step. These particular errors arise from a mismatch in punctuation and tokenization between the corpus used to train the parser, the PennTree bank, and the corpus which is being parsed and passed on to the rule extractor. Therefore we changed the quotation marks, which appear quite often in the parallel corpora, to opening and closing quotation marks. We also added some PennTree bank style tokenization rules[1]. These rules split contractions such as *I'll*, *It's*, *Don't*, *Gonna*, *Commissioner's* in order to correctly separate the verbs, negation and possessives that are

[1]The PennTree bank tokenization rules considered were taken from `http://www.cis.upenn.edu/~treebank/tokenizer.sed`. Further examples of contractions were added.

	Grammar Size		BLEU			
Parameters	Full	Filtered	2009-40	2010-40	2011-40	Average
Depth=3, Nodes=15, Size=3	2,572,222	751,355	18.57	20.43	18.51	19.17
Depth=4, Nodes=20, Size=4	3,188,970	901,710	18.88	20.38	18.63	19.30
Depth=5, Nodes=20, Size=5	3,668,205	980,057	19.04	20.47	18.75	**19.42**
Depth=5, Nodes=30, Size=5	3,776,961	980,061	18.90	20.59	18.77	19.42
Depth=5, Nodes=30, Size=6	4,340,716	1,006,174	18.98	20.52	18.80	19.43

Table 3: Cased BLEU scores for various rule extraction parameter settings for German-English language pair. The parameters considered are *MaxRuleDepth*, *MaxRuleSize*, *MaxNodes*. Grammar sizes are given for the full extracted grammar and after filtering for the newstest2008 dev set.

		newstest2012			newstest2013		
System	Sentences	BLEU	Glue Rule	Tree Depth	BLEU	Glue Rule	Tree Depth
Baseline	5,771	23.21	5.42	4.03	26.27	4.23	3.80
Big tuning set	10,068	23.52	3.41	4.34	26.33	2.49	4.03
Filtered tuning set	2,400	**23.54**	3.21	4.37	**26.30**	2.37	4.05

Table 4: Cased BLEU scores for German-English systems tuned on different data. Scores are emphasized for the system submitted to the shared translation task.

parsed as separate constituents.

For German–English, we carried out the usual compound splitting (Koehn and Knight, 2003), but not pre-reordering (Collins et al., 2005).

3.3 Rule Extraction

Some preliminary experiments were carried out for the German-English language pair to determine the parameters for the rule extraction step: *MaxRuleDepth*, *MaxRuleSize*, *MaxNodes*. Table 3 shows the BLEU score on different test sets for various parameter settings. For efficiency reasons less training data was used, therefore the grammar sizes, measured as the total number of extracted rules, are smaller than the final systems (Table 1). The parameters on the third line *Depth=5, Nodes=20, Size=4* were chosen as the average BLEU score did not increase although the size of the extracted grammar kept growing. Comparing the rate of growth of the full grammar and the grammar after filtering for the dev set (the columns headed "Full" and "Filtered") suggests that beyond this point not many more usable rules are extracted, even while the total number of rules stills increases.

3.4 Decoder Settings

We used the following non-default decoder parameters:

max-chart-span=25: This limits sub derivations to a maximum span of 25 source words. Glue rules are used to combine sub derivations allowing the full sentence to be covered.

ttable-limit=200: Moses prunes the translation grammar on loading, removing low scoring rules. This option increases the number of translation rules that are retained for any given source side RHS_s.

cube-pruning-pop-limit=1000: Number of hypotheses created for each chart span.

3.5 Tuning sets

One major limitation for the syntax-based systems is that decoding becomes inefficient for long sentences. Therefore using large tuning sets will slow down considerably the development cycle. We carried out some preliminary experiments to determine how the size of the tuning set affects the quality and speed of the system.

Three tuning sets were considered. The tuning set that was used for training the baseline system was built using the data from newstest2008-2010 filtering out sentences longer than 30 words. The second tuning set was built using all data from newstest2008-2011. The final tuning set was also built using the concatenation of the sets newstest2008-2011. All sentences in this set were decoded with a baseline system and the output was scored according to sentence-BLEU scores. We se-

lected examples with high sentence-BLEU score in a way that penalizes excessively short examples[2]. Results of these experiments are shown in Table 4.

Results show that there is some gain in BLEU score when providing longer sentences during tuning. Further experiments should consider tuning the baseline with the newstest2008-2011 data, to eliminate variance caused by having different data sources. Although the size of the third tuning set is much smaller than that of the other tuning sets, the BLEU score remains the same as when using the largest tuning set. The *glue rule* number, which shows how many times the glue rule was applied, is lowest when tuning with the third data set. The *tree depth* number, which shows the depth of the resulting target parse tree, is higher for the third tuning set as compared to the baseline and similar to that resulted from using the largest tuning set. These numbers are all indicators of better utilisation of the syntactic structure.

Regarding efficiency, the baseline tuning set and the filtered tuning set took about a third of the time needed to decode the larger tuning set.

Therefore we could draw some initial conclusions that providing longer sentences is useful, but sentences for which some baseline system performs very poorly in terms of BLEU score can be eliminated from the tuning set.

3.6 Results

Table 5 summarizes the results for the systems submitted to the shared task. The BLEU scores for the phrase-based system submitted by the University of Edinburgh are also shown for comparison. The syntax-based system had BLEU scores similar to those of the phrase-based system for German-English and English-German language pairs. For the Czech-English and Russian-English language pairs the syntax-based system was 2 BLEU points behind the phrase-based system.

However, in the manual evaluation, the German–English and English–German syntax based systems were ranked higher than the phrase-based systems. For Czech–English, the syntax systems also came much closer than the BLEU score would have indicated.

The Russian-English system performed worse because we used much less of the available data for training (leaving out *Commoncrawl*) and therefore the extracted grammar is less reliable. Another reason was the mismatch in data formatting for the Russian-English parallel corpus. All the training data was lowercased which resulted in more parsing errors.

[2]Ongoing work by Eva Hasler. Filtered data set was provided in order to speed up experiment cycles.

	phrase-based		syntax-based	
	BLEU	manual	BLEU	manual
en-de	**20.1**	0.571	19.4	**0.614**
de-en	**26.6**	0.586	26.3	**0.608**
cs-en	**26.2**	**0.562**	24.4	0.542
ru-en	**24.3**	**0.507**	22.5	0.416

Table 5: Cased BLEU scores and manual evaluation scores ("expected wins") on the newstest2013 evaluation set for the phrase-based and syntax-based systems submitted by the University of Edinburgh.

3.7 Post-Submission Experiments

Table 6 shows results for some preliminary experiments carried out for the German-English language pair that were not included in the final submission. The baseline system is trained on all available parallel data and tuned on data from newstest2008-2010 filtered for sentences up to 30 words.

Tree restructuring — In one experiment the parse trees were restructured before training by *left binarization*. Tree restructuring is need to improve generalization power of rules extracted from flat structures such as base noun phrases with several children. The second raw in Table 6 shows that the BLEU score did not improve and more glue rules were applied when using *left binarization*. One reason for this result is that the rule extraction parameters *MaxRuleDepth*, *MaxRuleSize*, *MaxNodes* had the same values as in the baseline. Increasing this parameters should improve the extracted grammar since binarizing the trees will increase these three dimensions.

Verb dropping — A serious problem of German–English machine translation is the tendency to drop verbs, which shatters sentence structure. One cause of this problem is the failure of the IBM Models to properly align the German verb to its English equivalent, since it is often dislocated with respect to English word order. Further problems appear when the main verb is not reordered in the target sentence, which can result in lower lan-

		newstest2012			newstest2013		
System	Grammar size	BLEU	glue rule	tree depth	BLEU	glue rule	tree depth
Baseline	55,310,162	23.21	5.42	4.03	26.27	4.23	3.80
Left binarized	57,151,032	23.17	7.79	4.09	26.13	6.57	3.85
Realigned vb	53,894,112	23.26	4.88	4.19	26.26	3.73	3.96

Table 6: Cased BLEU scores for various German-English systems.

System	Vb drop rules	Vb Count nt2012	Vb Count nt2013
Baseline	1,038,597	9,216	8,418
Realigned verbs	391,231	9,471	8,614
Reference translation	-	9,992	9,207

Table 7: Statistics about verb dropping.

guage model scores and BLEU scores. However the syntax models handle the reordering of verbs better than phrase-based models.

In an experiment we investigated how the number of verbs dropped by the translation rules can be reduced. In order to reduce the number of verb dropping rules we looked at unaligned verbs and realigned them before rule extraction. An unaligned verb in the source sentence was aligned to the verb in the target sentence for which IBM model 1 predicted the highest translation probability. The third row in Table 6 shows the results of this experiment. While there is no change in BLEU score the number of glue rules applied is lower. Further analysis shows in Table 7 that the number of verb dropping rules in the grammar is almost three times lower and that there are more translated verbs in the output when realigning verbs.

4 Conclusion

We describe in detail the syntax-based machine translation systems that we developed for four European language pairs. We achieved competitive results, especially for the language pairs involving German.

Acknowledgments

The research leading to these results has received funding from the European Union Seventh Framework Programme (FP7/2007-2013) under grant agreement 287658 (EU BRIDGE) and grant agreement 288487 (MosesCore).

References

Chen, S. F. and Goodman, J. (1998). An empirical study of smoothing techniques for language modeling. Technical report, Harvard University.

Collins, M., Koehn, P., and Kucerova, I. (2005). Clause restructuring for statistical machine translation. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL'05)*, pages 531–540, Ann Arbor, Michigan. Association for Computational Linguistics.

DeNeefe, S., Knight, K., Wang, W., and Marcu, D. (2007). What can syntax-based MT learn from phrase-based MT? In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL 2007). June 28-30, 2007. Prague, Czech Republic.*

Galley, M., Graehl, J., Knight, K., Marcu, D., DeNeefe, S., Wang, W., and Thayer, I. (2006). Scalable inference and training of context-rich syntactic translation models. In *ACL-44: Proceedings of the 21st International Conference on Computational Linguistics and the 44th annual meeting of the Association for Computational Linguistics*, pages 961–968, Morristown, NJ, USA. Association for Computational Linguistics.

Galley, M., Hopkins, M., Knight, K., and Marcu, D. (2004). What's in a translation rule? In *HLT-NAACL '04*.

Gao, Q. and Vogel, S. (2008). Parallel implementations of word alignment tool. In *Software Engineering, Testing, and Quality Assurance for Natural Language Processing*, SETQA-NLP '08, pages 49–57, Stroudsburg, PA, USA. Association for Computational Linguistics.

Hopkins, M. and Langmead, G. (2010). SCFG decoding without binarization. In *Proceedings of the 2010 Conference on Empirical Methods*

in Natural Language Processing, pages 646–655, Cambridge, MA. Association for Computational Linguistics.

Koehn, P. and Knight, K. (2003). Empirical methods for compound splitting. In *Proceedings of Meeting of the European Chapter of the Association of Computational Linguistics (EACL)*.

Koehn, P., Och, F. J., and Marcu, D. (2003). Statistical phrase-based translation. In *NAACL '03: Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology*, pages 48–54, Morristown, NJ, USA. Association for Computational Linguistics.

Marcu, D., Wang, W., Echihabi, A., and Knight, K. (2006). SPMT: statistical machine translation with syntactified target language phrases. In *EMNLP '06: Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing*, pages 44–52, Morristown, NJ, USA. Association for Computational Linguistics.

Och, F. J. (2003). Minimum error rate training in statistical machine translation. In *Proceedings of the 41st Annual Meeting on Association for Computational Linguistics - Volume 1*, ACL '03, pages 160–167, Morristown, NJ, USA. Association for Computational Linguistics.

Petrov, S., Barrett, L., Thibaux, R., and Klein, D. (2006). Learning accurate, compact, and interpretable tree annotation. In *Proceedings of the 21st International Conference on Computational Linguistics and the 44th annual meeting of the Association for Computational Linguistics*, ACL-44, pages 433–440, Stroudsburg, PA, USA. Association for Computational Linguistics.

Schmid, H. (2004). Efficient parsing of highly ambiguous context-free grammars with bit vectors. In *Proceedings of the 20th international conference on Computational Linguistics*, COLING '04, Stroudsburg, PA, USA. Association for Computational Linguistics.

Skut, W., Krenn, B., Brants, T., and Uszkoreit, H. (1997). An annotation scheme for free word order languages. In *Proceedings of the Fifth Conference on Applied Natural Language Processing (ANLP-97)*.

Stolcke, A. (2002). SRILM - an extensible language modeling toolkit. In *Intl. Conf. Spoken Language Processing, Denver, Colorado, September 2002*.

Wang, W., May, J., Knight, K., and Marcu, D. (2010). Re-structuring, re-labeling, and re-aligning for syntax-based machine translation. *Comput. Linguist.*, 36(2):247–277.

Williams, P. and Koehn, P. (2011). Agreement constraints for statistical machine translation into german. In *Proceedings of the Sixth Workshop on Statistical Machine Translation*, pages 217–226, Edinburgh, Scotland. Association for Computational Linguistics.

Williams, P. and Koehn, P. (2012). Ghkm rule extraction and scope-3 parsing in moses. In *Proceedings of the Seventh Workshop on Statistical Machine Translation*, pages 388–394, Montréal, Canada. Association for Computational Linguistics.