The RWTH Aachen German to English MT System for IWSLT 2015

Jan-Thorsten Peter, Farzad Toutounchi, Stephan Peitz, Parnia Bahar, Andreas Guta and Hermann Ney

Human Language Technology and Pattern Recognition Group
Computer Science Department
RWTH Aachen University
Aachen, Germany
<surname>@cs.rwth-aachen.de

Abstract

This work describes the statistical machine translation (SMT) systems of RWTH Aachen University developed for the evaluation campaign of the *International Workshop on Spoken Language Translation* (IWSLT) 2015. We participated in the MT and SLT tracks for the German→English language pair. We employ our state-of-the-art phrase-based and hierarchical phrase-based baseline systems for the MT track. The phrase-based system is augmented with joint translation and reordering model and maximum expected BLEU training for phrasal, lexical and reordering models. Furthermore, we apply feed-forward and recurrent neural language and translation models for reranking. We also train attention-based neural network models and utilize them in reranking the *n*-best lists for both phrase-based and hierarchical setups. On top of all our systems, we use system combination to enhance the translation quality by combining individually trained systems. In the SLT track, we additionally perform punctuation prediction on the automatic transcriptions employing hierarchical phrase-based translation.

1. Introduction

We describe the statistical machine translation (SMT) systems developed by RWTH Aachen University for the evaluation campaign of IWSLT 2015. We participated in the machine translation (MT) track and the spoken language translation (SLT) track for the German→English language pair. A combination of several single machine translation engines has proven to be highly effective on previous joint submission, e.g. [1, 2], and a similar approach is used for this task. We train individual systems using state-of-the-art phrase-based and hierarchical phrase-based translation engines. Each single system is a pipeline including either a phrase-based or a hierarchical decoder with additional models such as hierarchical reordering models, word class (cluster) language models, joint translation and reordering models, discriminative phrase training and reranking with different neural network models. For the spoken language translation task, the ASR output is enriched with punctuation and case information. The enrichment is performed by a hierarchical phrase-based translation system.

This paper is organized as follows. In Sections 2.1 through 2.3 we describe our translation software and baseline setups. Sections 2.4 and 2.5 provide further details about our joint translation and reordering models and discriminative phrase training, and sections 2.6, 2.7, and 2.8 describe the neural network models used in our translation systems, which are very effective in the shared task. Sections 2.9 explains the system combination pipeline applied on the individual systems for obtaining the combined system. Our experiments for each track are summarized in Section 3 and we conclude with Section 4.

2. SMT Systems

For the IWSLT 2015 evaluation campaign, RWTH utilizes state-of-the-art phrase-based and hierarchical translation systems. GIZA++ [3] is employed to train word alignments. We used *MultEval* [4] to evaluate our systems on the BLEU [5] and TER [6] measures. Due to using *MultEval*, BLEU scores are case-sensitive and TER scores are case-insensitive.

2.1. Phrase-based Systems

Our phrase-based decoder is the implementation of the *source cardinality synchronous search* (SCSS) procedure described in [7] in RWTH's open-source SMT toolkit, Jane 2.3[1] [8], which is freely available for non-commercial use. We use the standard set of models with phrase translation probabilities and lexical smoothing in both directions, word and phrase penalty, distance-based reordering model, *n*-gram target language models and enhanced low frequency feature [9]. The parameter weights are optimized with MERT [10] towards the BLEU metric. Additionally, we make use of a hierarchical reordering model (HRM) [11], a high-order word class language model (wcLM) [12], a joint translation and reordering model (cf. Section 2.4), a maximum expected BLEU training scheme (cf. Section 2.5) and reranking with different neural network models (cf. Sections 2.6, 2.7 and 2.8).

[1] http://www-i6.informatik.rwth-aachen.de/jane/

2.2. Hierarchical Phrase-based System

For our hierarchical setups, we also employ the open source translation toolkit Jane 2.3 [13]. In hierarchical phrase-based translation [14], a weighted synchronous context-free grammar is induced from parallel text. In addition to contiguous *lexical* phrases, *hierarchical* phrases with up to two gaps are extracted. The search is carried out with a parsing-based procedure. The standard models integrated into our Jane systems are phrase translation probabilities and lexical smoothing probabilities in both translation directions, word and phrase penalty, binary features marking hierarchical phrases, glue rule, and rules with non-terminals at the boundaries, enhanced low frequency feature and *n*-gram language models. We utilize the cube pruning algorithm [15] for decoding. Reranking the *n*-best lists using neural network models is also employed for our hierarchical systems.

2.3. Backoff Language Models

Both phrase-based and hierarchical translation systems use three backoff language models that are estimated with the KenLM toolkit [16] and are integrated into the decoder as separate models in the log-linear combination: A large general domain 5-gram LM, an in-domain 5-gram LM and a 7-gram word class language model (wcLM). All of them use interpolated Kneser-Ney smoothing. For the general domain LM, we first select $\frac{1}{2}$ of the English Shuffled News, French Shuffled News and both the English and French Gigaword corpora by the cross-entropy difference criterion described in [17]. The selection is then concatenated with all available remaining monolingual data and used to build an unpruned language model. The in-domain language model is estimated on the TED data only. For the word class LM, we train 200 classes on the target side of the bilingual training data using an in-house tool similar to `mkcls`. With these class definitions, we apply the technique shown in [12] to compute the wcLM on the same data as the general-domain LM.

2.4. Joint Translation and Reordering Models in Phrase-Based System

Joint translation and reordering (JTR) model [18] is introduced into the log-linear framework of our phrase-based system in order to include lexical and reordering dependencies beyond phrase-boundaries. The JTR model allows for more context than the previously developed extended translation model [19]. The unique JTR sequences are obtained by converting the full bilingual data and the corresponding Viterbi alignments. We train count-based 7-gram models with modified Kneser-Ney smoothing [20] on the JTR sequences using the KenLM toolkit [16].

In order to have the necessary information about the JTR sequences available during decoding, we annotate each phrase-table entry with the corresponding JTR sequence. Within the phrase-based decoder, we extend each search state such that it additionally stores the JTR model history. During decoding, a reordering token has to be appended to the beginning of the hypothesized JTR sequence, if the alignment step from the previous JTR token in the history to the current token is non-monotone.

Including the JTR model improved our phrase-based baseline system by 0.7 BLEU on tst2013.

2.5. Maximum Expected BLEU Training

Discriminative training is a powerful method to learn a large number of features with respect to a given error metric. In this work we learn two types of features under a maximum expected BLEU objective [21]. We used the TED portion of the data for discriminative training, since it is high quality in-domain data of reasonable size. This makes training feasible while at the same time providing an implicit domain adaptation effect. For our gradient based update method we generate 100-best lists on the training data which are used as training samples similar to [21]. A leave-one-out heuristic [22] is applied to circumvent over-fitting. Here, we follow an approach similar to [23]. Each feature type is first discriminatively trained, then condensed into a single feature for the log-linear model combination and finally optimized with MERT. We simultaneously train phrase pair features and phrase-internal word pair features, adding two models to the log-linear combination. In the tables in Section 3 we denote the maximum expected BLEU training as *MaxExpBleu*.

2.6. Feed-Forward Neural Network Models

We use four feed-forward neural network (FFNN) models with similar structure as the models used by [24, 25]. The models and following neural network models are applied for reranking 1000-best lists. The new weights are trained with one additional MERT iteration.

All networks are trained with different input features or layers:

- Language model (LM), the 7 last words on the target side, with two hidden layers (1000 and 500 nodes)
- Joint model (JM), the 5 source words around the aligned source word (2 before the aligned word, and 2 after it) and the 4 last words on the target side, with two hidden layers (1000 and 500 nodes)
- Translation model (TM), the 5 source words around the aligned source word, with two hidden layers (1000 and 500 nodes)
- Translation model (TM), the 5 source words around the aligned source word, with three hidden layers (2000, 2000, and 1000 nodes)

The output layer in all cases is a softmax layer with a short list of 10000. All remaining words are clustered into 1000 classes, and the corresponding class probabilities are predicted. The neural network was implemented using Theano [26, 27].

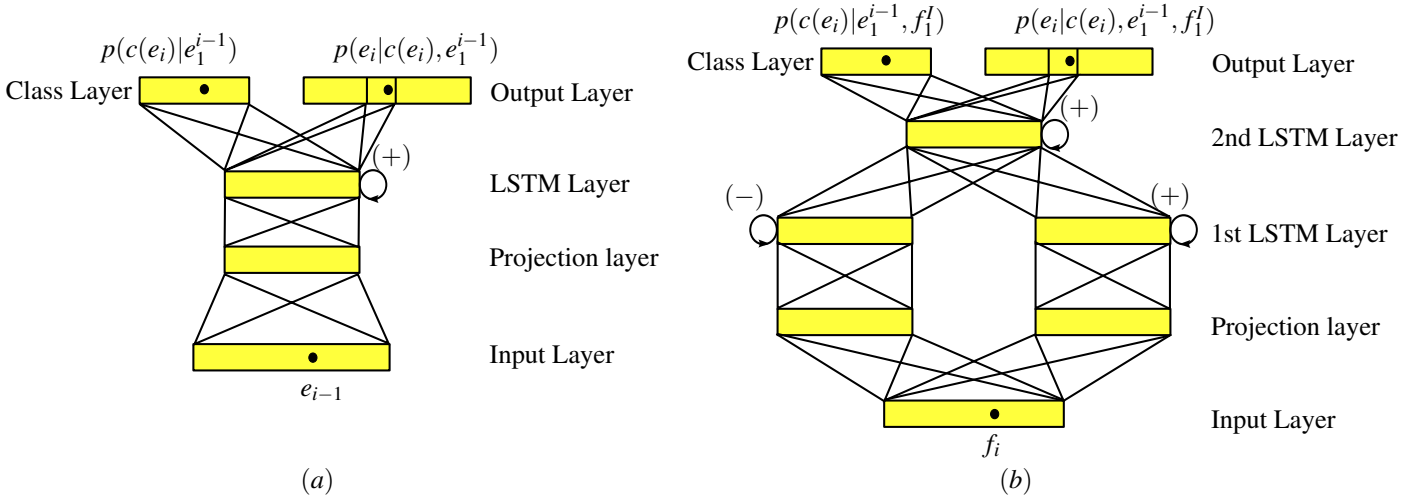

Figure 1: Architecture of the deep recurrent (a) language model, and (b) bidirectional translation model. By $(+)$ and $(-)$, we indicate a processing in forward and backward time directions, respectively. A single source projection matrix is used for the forward and backward branches.

2.7. Recurrent Neural Network Models

Our systems apply reranking on 1000-best lists using recurrent language and translation models. The recurrency is handled with the long short-term memory (LSTM) architecture [28] and we use a class-factored output layer for increased efficiency as described in [29]. All neural networks are trained using 2000 word classes. In addition to the recurrent language model (RNN-LM), we apply the deep bidirectional word-based translation model (RNN-BTM) described in [30]. This requires a one-to-one word alignment, which is generated by introducing ε tokens and using an IBM1 translation table. We apply the *bidirectional* version of the translation model, which uses both forward and backward recurrency in order to take the *full source context* into account for each translation decision. Two language models are used for reranking, one is trained on the in-domain data, and the other on the entire monolingual data. The in-domain language model is set up with 300 nodes in both the projection and the hidden LSTM layer, while the general-domain language model is set up with 500 nodes in both layers. The general-domain language model is the same model which was used in the IWSLT 2014 evaluations [31]. For the BTM, the in-domain bilingual data is used for training. Furthermore, we use 200 nodes in all layers, namely the forward and backward projection layers, the first hidden layers for both forward and backward processing and the second hidden layer, which joins the output of the directional hidden layers. The architecture of the LM and BTM networks are shown in Figure 1. The neural network was implemented using the RWTHLM toolkit.[2]

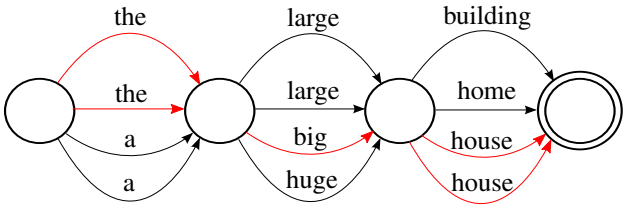

Figure 2: System A: *the large building*; System B: *the large home*; System C: *a big house*; System D: *a huge house*; Reference: *the big house*.

2.8. Attention Based Recurrent Neural Network

The neural network models described in Section 2.6 and Section 2.7 are either used as pure language models or rely on the alignments given by the underlying system. To avoid this dependency on the alignment while maintaining the translation model we also use an attention-based recurrent neural network model as proposed in [32]. The model uses gated recurrent units as proposed by [33]. They have comparable properties to the LSTM architecture used by the recurrent neural networks in Section 2.7. We use a bidirectional layer on the source side with 1000 nodes for each direction and a unidirectional model with 1000 nodes for the target side. The GroundHog toolkit[3] was used to train two models, one on the in-domain data and one on the full data.

2.9. System Combination

System combination is applied to produce consensus translations from multiple hypotheses which are obtained from different translation approaches. The consensus translations outperform the individual hypotheses in terms of translation quality. A system combination implementation which has been developed at RWTH Aachen University [34] is used to

[2] https://www-i6.informatik.rwth-aachen.de/web/Software/rwthlm.php

[3] https://github.com/lisa-groundhog/GroundHog

combine the outputs of different engines.

The first step in system combination is generation of confusion networks (CN) from I input translation hypotheses. We need pairwise alignments between the input hypotheses, and the alignments are obtained by METEOR [35]. The hypotheses are then reordered to match a selected skeleton hypothesis in terms of word ordering. We generate I different CNs, each having one of the input systems as the skeleton hypothesis, and the final lattice will be the union of all I generated CNs. In Figure 2 an example of a confusion network with $I = 4$ input translations is depicted. The decoding of a confusion network is finding the shortest path in the network. Each arc is assigned a score of a linear model combination of M different models, which include word penalty, 3-gram language model trained on the input hypotheses, a binary primary system feature that marks the primary hypothesis, and a binary voting feature for each system. The binary voting feature for a system is 1 iff the decoded word is from that system, otherwise 0. The different model weights for system combination are trained with MERT.

3. Experimental Evaluation

3.1. Machine Translation (MT) Track

For the German→English machine translation task, the word alignment is trained with GIZA++ and we apply the phrase-based decoder, as well as the hierarchical phrase-based decoder implemented in Jane. We use all permissible parallel data for the IWSLT 2015 systems in training the translation model. In a preprocessing step the German source is decompounded [36] and part-of-speech-based long-range verb reordering rules [37] are applied. The baseline contains three backoff language models, namely a general-domain LM, an in-domain LM and a word class LM as described in Section 2.3, and the hierarchical reordering model (HRM). In addition, we tune our systems on the development set `dev2012`, which contains manual transcriptions from German talks and is more similar to the evaluation data. As `tst2013` is also a manual transcription of TED talks, we will focus on the results for the `dev2012`-tuned system on this evaluation data set. The performance of the individual MT systems based on phrase-based and hierarchical phrase-based decoders is summarized in Table 1.

The phrase-based baseline reaches a performance of 28.0 BLEU on `tst2013`. Adding the joint translation and reordering (JTR) models to baseline increases the BLEU scores to 28.7 on `tst2013`. Introducing maximum expected BLEU training on top of JTR improves the translation quality by 0.5 BLEU on `tst2013`. We also apply different neural network models for reranking the 1000-best lists obtained by phrase-based system which is augmented with JTR. We use the four feed-forward models described in Section 2.6, and they each improve the JTR system by 0.1 to 0.3 BLEU. Moreover, we employ recurrent models described in Section 2.7, and depending on the model they can also improve the performance by up to 0.4 BLEU. Introducing the attention-based recurrent model (cf. Section 2.8), enhances the translation quality of the phrase-based system with JTR by 0.8 BLEU. So far all the neural network models were applied individually. In the last two rows of the phrase-based section in Table 1, we use all the above neural networks simultaneously for reranking the n-best lists of the phrase-based system including JTR, and we improve the translation quality by 1.1 and 1.2 BLEU on `tst2013` in two different optimization runs.

The hierarchical baseline system reaches a performance of 28.8 BLEU on `tst2013`. We tried to add source reordering to the hierarchical baseline. Athough it does not improve the translation quality of `tst2013`, we keep it as an individual system for our system combination pipeline. Applying a feed-forward neural network language model and a recurrent neural network language model for reranking the 1000-best lists obtained by hierarchical baseline system improves the translation quality by 0.1 and 0.2 BLEU, respectively. We also use the attention-based recurrent neural network in reranking, and it boosts the BLEU scores by 1 and 1.2 points in two different optimization runs. Using attention-based networks trained on the in-domain data also enhances the translation quality of baseline by 0.5 BLEU. Furthermore, we use all the above neural networks at the same time for reranking the n-best lists of the hierarchical baseline system, and the improvement on `tst2013` is 1.1 BLEU.

The final submission system for the MT track of IWSLT 2015 German→English task is the combination of all single systems in Table 1 using the methods described in Section 2.9. In total, 20 systems are combined, and the parameters are tuned on `dev2012`. The performance of the combined system is summarized in Table 2. Comparing to our 2014 submission system, we have an improvement of 1.2 BLEU on `tst2014`.

3.2. Spoken Language Translation (SLT) Track

RWTH participated in the German→English SLT task. Punctuation marks and case information are reintroduced by applying a monolingual hierarchical phrase-based translation system as described in [38]. In such a system, hierarchical phrases with a maximum of one non-terminal symbol are extracted and the feature weights can be tuned with MERT. In addition, we add a word class language model (wcLM) to the log-linear model combination.

Table 3 shows a comparison of monolingual phrase-based [39] and hierarchical translation systems tuned on different optimization criteria.

For this task, tuning a monolingual hierarchical translation system on BLEU seems to work better than optimizing towards F_2-Score. In any case it outperforms the phrase-based system. Furthermore, applying a word class language model (wcLM) seems to help as well in terms of BLEU and TER.

Since punctuation prediction and recasing are applied before the actual translation, our translation system can be kept

Table 1: Results of the individual systems for the German→English MT task. BLEU scores are case-sensitive and TER scores are case-insensitive.

	dev2012		tst2010		tst2011		tst2012		tst2013	
Individual Systems	BLEU	TER	BLEU	TER	BLEU	TER	BLEU	TER	BLEU	TER
SCSS Baseline	25.3	59.6	29.8	48.8	35.4	43.4	29.7	48.9	28.0	51.1
+ **JTR model**	26.4	58.6	30.0	48.4	36.2	42.6	30.3	48.1	28.7	50.3
+ MaxExpBleu	26.8	57.7	30.9	47.2	37.0	41.8	30.7	47.5	29.2	49.9
+ FFNN-LM	26.6	58.3	30.4	48.4	36.6	42.4	30.6	48.3	29.0	50.1
+ FFNN-JM	26.7	58.2	30.1	48.4	36.3	42.4	30.7	48.0	28.8	50.4
+ FFNN-TM	26.7	58.2	30.1	48.4	36.4	42.2	30.4	48.1	28.9	50.1
+ FFNN-TM*	26.4	58.3	31.3	47.6	37.4	41.7	31.6	47.3	29.0	50.1
+ RNN-LM	26.8	58.3	30.0	48.4	36.1	42.6	30.4	48.3	29.1	50.0
+ RNN-LM-InDomain	26.3	58.4	30.4	48.3	36.6	42.3	30.5	48.1	28.2	50.9
+ RNN-BTM	26.7	57.8	30.8	47.7	37.3	41.7	31.2	47.2	29.1	49.9
+ RNN-Attention	27.0	57.9	31.5	47.1	38.0	41.2	31.8	46.8	29.5	49.6
+ AllAboveNNs	27.4	57.1	31.2	47.2	36.7	42.0	31.6	47.2	29.9	49.0
+ AllAboveNNs†	27.9	56.5	31.8	46.5	37.6	41.1	31.5	46.7	29.8	48.9
Hierarchical Baseline	25.3	60.0	30.2	49.3	35.3	44.0	30.1	49.0	28.8	51.6
+ SrcReordering	25.7	59.2	30.0	49.1	35.7	43.6	30.0	48.9	28.4	51.1
+ FFNN-LM	25.4	60.3	30.1	49.4	35.5	43.8	30.0	49.3	28.9	51.7
+ RNN-LM	25.9	60.0	29.9	49.4	35.2	43.7	30.1	49.2	29.0	51.4
+ RNN-Attention	26.4	59.3	31.4	48.2	36.5	42.9	31.4	47.8	30.0	50.5
+ RNN-Attention†	26.4	59.3	30.6	48.9	35.8	43.6	30.6	48.6	29.8	50.6
+ RNN-Attention-InDomain	26.3	59.0	30.8	48.5	36.0	43.2	30.8	48.3	29.3	50.9
+ AllAboveNNs	26.7	58.6	31.9	47.6	36.9	42.4	31.8	47.3	29.9	50.4

* This FFNN-TM has three hidden layers. The other FFNNs have two hidden layers. (cf. Section 2.6)

† A different optimization run.

Table 2: Results of the combined system for the German→English MT task submission. `tst2014` and `tst2015` results are computed by the task organizers. BLEU scores are case-sensitive and TER scores are case-insensitive.

	dev2012		tst2013		tst2014		tst2015	
System	BLEU	TER	BLEU	TER	BLEU	TER	BLEU	TER
Best Individual System	27.4	57.1	29.9	49.0	25.2	56.4	31.1	48.3
Combined System (2015 Submission)	28.2	57.0	30.5	49.0	26.2	55.2	31.5	47.1
2014 Submission	27.0	57.2	27.6	52.1	25.0	55.5	-	-

Table 3: Results of the German→English SLT task. Scores for `tst2015` (case-sensitive) are computed by the task organizers.

	Prediction	**Optimization**	dev2012		tst2013		tst2015	
System	**Method**	**Criterion**	BLEU	TER	BLEU	TER	BLEU	TER
SCSS Baseline	phrase-based	F_2	20.5	62.6	18.6	63.7	-	-
		BLEU	20.0	65.1	18.4	65.8	-	-
	hierarchical	F_2	20.9	62.1	18.7	63.4	-	-
		BLEU	20.9	62.5	19.0	63.4	-	-
	+ wcLM	BLEU	21.3	61.7	19.1	62.8	-	-
+ AllAboveNNs	+ wcLM	BLEU	21.6	61.1	19.8	62.4	18.8	65.2

completely unchanged and we are able to use our final system from the MT track directly. We use *SCSS Baseline + AllAboveNNs* (cf. Table 1) for our final submission.

4. Conclusion

RWTH participated in the MT and SLT tracks for the German→English IWSLT 2015 evaluation campaign.

The baseline systems for the MT track utilize our state-of-the-art phrase-based and hierarchical translation decoders and we were able to improve them by applying maximum expexted BLEU training and employing several neural network models for reranking the *n*-best lists. We built several single machine translation engines which are based on either phrase-based or hierarchical decoders, and combined all the built systems using our system combination pipeline. We achieve a performance of 26.2 in BLEU and 55.2 in TER for `tst2014` and 31.5 in BLEU and 47.1 in TER for `tst2015`, and we improve the BLEU scores by 1.2 point on the `tst2014` compared to our 2014 system.

For the SLT track, the ASR output was enriched with punctuation and casing information by a hierarchical translation system.

5. Acknowledgements

This paper has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement n° 645452 (QT21).

6. References

[1] M. Freitag, S. Peitz, J. Wuebker, H. Ney, N. Durrani, M. Huck, P. Koehn, T.-L. Ha, J. Niehues, M. Mediani, T. Herrmann, A. Waibel, N. Bertoldi, M. Cetolo, and M. Federico, "EU-BRIDGE MT: Text Translation of Talks in the EU-BRIDGE Project," in *Proc. of the Int. Workshop on Spoken Language Translation (IWSLT)*, Heidelberg, Germany, Dec. 2013, pp. 128–135.

[2] M. Freitag, S. Peitz, J. Wuebker, H. Ney, M. Huck, R. Sennrich, N. Durrani, M. Nadejde, P. Williams, P. Koehn, T. Herrmann, E. Cho, and A. Waibel, "EU-BRIDGE MT: Combined Machine Translation," in *Proc. of the Workshop on Statistical Machine Translation (WMT)*, Baltimore, MD, USA, June 2014, pp. 105–113.

[3] F. J. Och and H. Ney, "A Systematic Comparison of Various Statistical Alignment Models," *Computational Linguistics*, vol. 29, no. 1, pp. 19–51, Mar. 2003.

[4] J. H. Clark, C. Dyer, A. Lavie, and N. A. Smith, "Better Hypothesis Testing for Statistical Machine Translation: Controlling for Optimizer Instability," in *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies: Short Papers - Volume 2*, ser. HLT '11. Stroudsburg, PA, USA: Association for Computational Linguistics, 2011, pp. 176–181.

[5] K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu, "Bleu: a Method for Automatic Evaluation of Machine Translation," in *Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics*, Philadelphia, Pennsylvania, USA, July 2002, pp. 311–318.

[6] M. Snover, B. Dorr, R. Schwartz, L. Micciulla, and J. Makhoul, "A Study of Translation Edit Rate with Targeted Human Annotation," in *Proceedings of the 7th Conference of the Association for Machine Translation in the Americas*, Cambridge, Massachusetts, USA, August 2006, pp. 223–231.

[7] R. Zens and H. Ney, "Improvements in Dynamic Programming Beam Search for Phrase-based Statistical Machine Translation," in *International Workshop on Spoken Language Translation*, Honolulu, Hawaii, Oct. 2008, pp. 195–205.

[8] J. Wuebker, M. Huck, S. Peitz, M. Nuhn, M. Freitag, J.-T. Peter, S. Mansour, and H. Ney, "Jane 2: Open source phrase-based and hierarchical statistical machine translation," in *International Conference on Computational Linguistics*, Mumbai, India, Dec. 2012, pp. 483–491.

[9] B. Chen, R. Kuhn, G. Foster, and H. Johnson, "Unpacking and Transforming Feature Functions: New Ways to Smooth Phrase Tables," in *MT Summit XIII*, Xiamen, China, Sept. 2011, pp. 269–275.

[10] F. J. Och, "Minimum Error Rate Training in Statistical Machine Translation," in *Proc. of the 41th Annual Meeting of the Association for Computational Linguistics (ACL)*, Sapporo, Japan, July 2003, pp. 160–167.

[11] M. Galley and C. D. Manning, "A simple and effective hierarchical phrase reordering model," in *Proceedings of the Conference on Empirical Methods in Natural Language Processing*. Stroudsburg, PA, USA: Association for Computational Linguistics, 2008, pp. 848–856.

[12] J. Wuebker, S. Peitz, F. Rietig, and H. Ney, "Improving statistical machine translation with word class models," in *Conference on Empirical Methods in Natural Language Processing*, Seattle, WA, USA, Oct. 2013, pp. 1377–1381.

[13] D. Vilar, D. Stein, M. Huck, and H. Ney, "Jane: Open source hierarchical translation, extended with reordering and lexicon models," in *ACL 2010 Joint Fifth Workshop on Statistical Machine Translation and Metrics MATR*, Uppsala, Sweden, July 2010, pp. 262–270.

[14] D. Chiang, "Hierarchical Phrase-Based Translation," *Computational Linguistics*, vol. 33, no. 2, pp. 201–228, 2007.

[15] L. Huang and D. Chiang, "Forest Rescoring: Faster Decoding with Integrated Language Models," in *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics*, Prague, Czech Republic, June 2007, pp. 144–151.

[16] K. Heafield, I. Pouzyrevsky, J. H. Clark, and P. Koehn, "Scalable modified Kneser-Ney language model estimation," in *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics*, Sofia, Bulgaria, August 2013, pp. 690–696.

[17] R. Moore and W. Lewis, "Intelligent Selection of Language Model Training Data," in *ACL (Short Papers)*, Uppsala, Sweden, July 2010, pp. 220–224.

[18] A. Guta, T. Alkhouli, J.-T. Peter, J. Wuebker, and H. Ney, "A Comparison between Count and Neural Network Models Based on Joint Translation and Reordering Sequences," in *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*. Lisbon, Portugal: Association for Computational Linguistics, Sept. 2015.

[19] A. Guta, J. Wuebker, M. Graça, Y. Kim, and H. Ney, "Extended Translation Models in Phrase-based Decoding," in *Proceedings of the EMNLP 2015 Tenth Workshop on Statistical Machine Translation*, Lisbon, Portugal, Sept. 2015.

[20] S. F. Chen and J. Goodman, "An Empirical Study of Smoothing Techniques for Language Modeling," Computer Science Group, Harvard University, Cambridge, MA, Tech. Rep. TR-10-98, Aug. 1998.

[21] X. He and L. Deng, "Maximum Expected BLEU Training of Phrase and Lexicon Translation Models," in *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (ACL)*, Jeju, Republic of Korea, Jul 2012, pp. 292–301.

[22] J. Wuebker, A. Mauser, and H. Ney, "Training phrase translation models with leaving-one-out," in *Proceedings of the 48th Annual Meeting of the Assoc. for Computational Linguistics*, Uppsala, Sweden, July 2010, pp. 475–484.

[23] M. Auli, M. Galley, and J. Gao, "Large Scale Expected BLEU Training of Phrase-based Reordering Models," in *Conference on Empirical Methods in Natural Language Processing (EMNLP)*, Doha, Qatar, Oct 2014.

[24] 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, June 2014, pp. 1370–1380.

[25] H.-S. Le, A. Allauzen, and F. Yvon, "Continuous Space Translation Models with Neural Networks," Montréal, Canada, June 2012, pp. 39–48.

[26] J. Bergstra, O. Breuleux, F. Bastien, P. Lamblin, R. Pascanu, G. Desjardins, J. Turian, D. Warde-Farley, and Y. Bengio, "Theano: a CPU and GPU math expression compiler," in *Proceedings of the Python for Scientific Computing Conference (SciPy)*, June 2010, oral Presentation.

[27] F. Bastien, P. Lamblin, R. Pascanu, J. Bergstra, I. J. Goodfellow, A. Bergeron, N. Bouchard, and Y. Bengio, "Theano: new features and speed improvements," Deep Learning and Unsupervised Feature Learning NIPS 2012 Workshop, 2012.

[28] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[29] M. Sundermeyer, R. Schlüter, and H. Ney, "LSTM Neural Networks for Language Modeling," in *Interspeech*, Portland, OR, USA, Sept. 2012.

[30] M. Sundermeyer, T. Alkhouli, J. Wuebker, and H. Ney, "Translation Modeling with Bidirectional Recurrent Neural Networks," in *Conference on Empirical Methods in Natural Language Processing*, Doha, Qatar, Oct. 2014, pp. 14–25.

[31] J. Wuebker, S. Peitz, A. Guta, and H. Ney, "The RWTH Aachen Machine Translation Systems for IWSLT 2014," in *International Workshop on Spoken Language Translation*, Lake Tahoe, CA, USA, Dec. 2014.

[32] D. Bahdanau, K. Cho, and Y. Bengio, "Neural Machine Translation by Jointly Learning to Align and Translate," in *Proceedings of the International Conference on Learning Representations (ICLR)*, San Diego, May 2015.

[33] K. Cho, B. van Merrienboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, "Learning Phrase Representations using RNN Encoder–Decoder for Statistical Machine Translation," in *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Doha, Qatar: Association for Computational Linguistics, October 2014, pp. 1724–1734.

[34] M. Freitag, M. Huck, and H. Ney, "Jane: Open Source Machine Translation System Combination," in *Proc. of*

the Conf. of the European Chapter of the Assoc. for Computational Linguistics (EACL), Gothenberg, Sweden, Apr. 2014, pp. 29–32.

[35] S. Banerjee and A. Lavie, "METEOR: An Automatic Metric for MT Evaluation with Improved Correlation with Human Judgments," in *43rd Annual Meeting of the Assoc. for Computational Linguistics: Proc. Workshop on Intrinsic and Extrinsic Evaluation Measures for MT and/or Summarization*, Ann Arbor, MI, June 2005, pp. 65–72.

[36] P. Koehn and K. Knight, "Empirical Methods for Compound Splitting," in *Proceedings of European Chapter of the ACL (EACL 2003)*, 2003, pp. 187–194.

[37] M. Popović and H. Ney, "POS-based Word Reorderings for Statistical Machine Translation," in *International Conference on Language Resources and Evaluation*, 2006, pp. 1278–1283.

[38] S. Peitz, M. Freitag, and H. Ney, "Better Punctuation Prediction with Hierarchical Phrase-Based Translation," in *International Workshop on Spoken Language Translation*, Lake Tahoe, CA, USA, Dec. 2014, pp. 271–278.

[39] S. Peitz, M. Freitag, A. Mauser, and H. Ney, "Modeling punctuation prediction as machine translation," in *International Workshop on Spoken Language Translation*, San Francisco, CA, USA, Dec. 2011, pp. 238–245.