The RWTH Aachen University English-German and German-English Machine Translation System for WMT 2017

Jan-Thorsten Peter, Andreas Guta, Tamer Alkhouli, Parnia Bahar,
Jan Rosendahl, Nick Rossenbach, Miguel Graça and Hermann Ney
Human Language Technology and Pattern Recognition Group
Computer Science Department
RWTH Aachen University
D-52056 Aachen, Germany
<surname>@i6.informatik.rwth-aachen.de

Abstract

This paper describes the statistical machine translation system developed at RWTH Aachen University for the English→German and German→English translation tasks of the *EMNLP 2017 Second Conference on Machine Translation* (WMT 2017). We use ensembles of attention-based neural machine translation system for both directions. We use the provided parallel and synthetic data to train the models. In addition, we also create a phrasal system using joint translation and reordering models in decoding and neural models in rescoring.

1 Introduction

We describe the statistical machine translation (SMT) systems developed by RWTH Aachen University for the German→English and English→German language pairs of the WMT 2017 evaluation campaign. After testing multiple systems and system combinations we submitted an ensemble of multiple NMT networks since it outperformed every tested system combination.

This paper is organized as follows. In Section 2 we describe our data preprocessing. Section 3 depicts the generation of synthetic data. Our translation software and baseline setups are explained in Section 4, including the attention-based recurrent neural network ensemble in Subsection 4.1 and phrasal joint translation and reordering (JTR) system in Subsection 4.2. Our experiments for each track are summarized in Section 5.

2 Preprocessing

We compared two different preprocessings for German→English for the attention-based recurrent neural network (NMT) system. The first preprocessing is similar to the preprocessing used in our WMT 2015 submission (Peter et al., 2015), which was optimized for phrase-based translation (PBT).

Secondly, we utilize a simplified version which uses tokenization, frequent casing, and simple categories only. Note, that the changes in preprocessing have a huge negative impact on the PBT system, while slightly improving the NMT system (Table 1). We therefore use the simplified version for all pure NMT experiments and use the old preprocessing for all other systems.

The phrasal JTR system uses the preprocessing technique that is optimized for PBT, as it relies on phrases as translation candidates. The preprocessing is similar to the one used in the WMT 2015 submission, but without any pre-ordering of source words. The English→German NMT system utilizes only the simplified preprocessing.

3 Synthetic Source Sentences

To increase the amount of usable parallel training data for the phrase-based and the neural machine translation systems, we translate a subset of the monolingual training data back to English in a similar way as described by (Bertoldi and Federico, 2009) and (Sennrich et al., 2016b).

We create a baseline German→English NMT system as described in 4.1 which is trained with all parallel data to translate 6.9M English sentences into German. For the other direction we use this newly created synthetic data and the parallel corpus to train a baseline English→German system, which in turn is used to translate additional 4.4M sentences from English to German.

Further, we append the synthetic data created by (Sennrich et al., 2016a). This results in additional 4.2M sentences for the German→English system and 3.6M for the opposite direction.

Proceedings of the Conference on Machine Translation (WMT), Volume 2: Shared Task Papers, pages 358–365
Copenhagen, Denmark, September 711, 2017.

Systems	PP	newstest2015 BLEU	TER	cTER	BEER	newstest2016 BLEU	TER	cTER	BEER	newstest2017 BLEU	TER	cTER	BEER
PBT	WMT15	27.9	52.7	53.9	60.5	33.6	47.8	49.1	63.5	28.9	52.2	54.3	60.8
PBT	simple	26.6	54.3	55.3	59.1	31.4	49.4	50.8	62.1	27.1	53.7	56.1	59.4
NMT	WMT15	27.3	53.0	52.7	59.7	32.1	48.4	48.4	62.8	27.7	53.0	53.0	59.9
NMT	simple	27.7	52.3	52.4	59.8	32.1	47.9	47.8	62.7	27.9	52.3	52.5	60.2

Table 1: Compares the performance of the preprocessing (PP) optimized for phrase-based systems (WMT15) or a very simple setup (simple), as described in Section 2 on a PBT and a Neural Machine Translation (NMT) system.

Individual Systems	newstest2015 BLEU	TER	cTER	BEER	newstest2016 BLEU	TER	cTER	BEER	newstest2017 BLEU	TER	cTER	BEER
Baseline	27.7	52.3	52.4	59.8	32.1	47.9	47.8	62.7	27.9	52.3	52.5	60.2
+ fertility	28.2	51.8	51.9	60.2	32.9	47.1	47.3	63.2	28.6	51.5	51.7	60.6
+ synthetic data	29.9	50.1	49.3	61.4	36.7	44.0	44.0	65.2	30.6	49.7	49.6	61.8
+ 2-layers decoder	30.7	49.7	48.3	61.8	37.5	43.6	43.4	65.5	31.8	49.1	49.1	62.3
+ filtered	30.8	49.7	48.5	61.8	37.9	43.1	42.8	65.8	31.7	49.1	48.9	62.2
+ anneling scheme	31.1	49.8	48.4	61.9	37.9	43.6	43.1	65.7	32.2	48.9	48.6	62.4
Base system	31.3	49.5	48.2	62.0	37.9	43.6	43.1	65.7	32.1	49.1	48.7	62.4
+ connected all LSTM cells	30.7	49.8	49.0	61.5	37.4	43.9	43.5	65.4	31.7	49.3	48.9	62.2
+ fertility	31.1	49.8	48.4	61.9	37.9	43.6	43.1	65.7	32.2	48.9	48.6	62.4
+ alignment feedback	31.3	49.8	48.3	61.9	37.7	43.6	43.2	65.6	32.2	49.1	48.4	62.4
Ensemble	32.0	48.9	47.5	62.3	38.8	42.7	42.5	66.2	33.1	48.3	47.7	63.0

Table 2: Results of the individual systems for the German→English task. The base system contains synthetic data, 2-decoder layers, filtered rapid data, and was trained with annealing learning rate instead of merging. Details are explained in Section 4.1.

4 SMT Systems

For the WMT 2017 evaluation campaign, we have employed two different translation system architectures for the German→English direction:

- phrasal joint translation and reordering
- attention-based neural network ensemble

The word alignments required by some models are obtained with GIZA++ (Och and Ney, 2003). We use mteval from the Moses toolkit (Koehn et al., 2007) and TERCom to evaluate our systems on the BLEU (Papineni et al., 2002) and TER (Snover et al., 2006) measures. Additional we use BEER (Stanojević and Sima'an, 2014) and CTER (Wang et al., 2016). All reported scores are case-sensitive and normalized.

4.1 Attention-Based Recurrent Neural Network

The best performing system provided by the RWTH is an attention-based recurrent neural network (NMT) similar to (Bahdanau et al., 2015). We use an implementation based on Blocks (van Merriënboer et al., 2015) and Theano (Bergstra et al., 2010; Bastien et al., 2012).

The encoder and decoder word embeddings are of size 620. The encoder consists of a bidirectional layer with 1000 LSTMs with peephole connections (Hochreiter and Schmidhuber, 1997a) to encode the source side. Additionally we ran experiments with two layers using 1000 LSTM nodes each where we optionally connect all internal states of the first LSTM layer to the second. The data is converted into subword units using byte pair encoding with 20000 operations (Sennrich et al., 2016c).

During training a batch size of 50 is used. The applied gradient algorithm is Adam (Kingma and Ba, 2014) with a learning rate of 0.001 and the four best models are averaged as described in the beginning of (Junczys-Dowmunt et al., 2016). Later experiments are done using Adam followed by an annealing scheme for learning rate reduction for SGD, as described in (Bahar et al., 2017).

The network is trained with 30% dropout for up to 500K iterations and evaluated every 10000 iterations on newstest2015. Decoding is done using a beam search with a beam size of 12.

If the neural network creates a special number token, the corresponding source number with

the highest attention weight is copied to the target side. The synthetic training data is created and used as described in Section 3.

In addition, we tested methods to provide the alignment computation with supplementary information comparable with (Tu et al., 2016; Cohn et al., 2016). We model the word fertility and feedback the information of the last alignment points using a conventional layer with a window size of 5.

The final system was an ensemble of multiple systems each trained with slightly different settings as shown in Table 2 and 4.

4.2 Phrasal Joint Translation and Reordering System

The phrasal Joint Translation and Reordering (JTR) decoder is based on the implementation of the *source cardinality synchronous search* (SCSS) procedure described in (Zens and Ney, 2008). The system combines the flexibility of word-level models with the search accuracy of phrase candidates. It incorporates the JTR model (Guta et al., 2015), a language model (LM), a word class language model (wcLM) (Wuebker et al., 2013), phrasal translation probabilities, conditional JTR probabilities on phrase level and additional lexical models for smoothing purposes. The phrases are annotated with word alignments to allow for the application of word-level models.

A more detailed description of the translation candidate generation and the search procedure is given in (Peter et al., 2016). The phrase extraction and the estimation of the translation models are performed on all bilingual data excluding the rapid2016 corpus, the newstest2008-2013 and newssyscom2009 corpora and the first part of the synthetic data (Section 3). The non-synthetic data was filtered to contain only sentences with 4 unaligned words at most. In total, this results in 3.57M parallel and 6.94M synthetic sentences.

4.2.1 JTR Model

A JTR sequence $(\tilde{f}, \tilde{e})_1^{\tilde{I}}$ is an interpretation of a bilingual sentence pair (f_1^J, e_1^I) and its word alignment b_1^I. The joint probability $p(f_1^J, e_1^I, b_1^I)$ can be modeled as:

$$\begin{aligned} p(f_1^J, e_1^I, b_1^I) &= p((\tilde{f}, \tilde{e})_1^{\tilde{I}}) \\ &= \prod_{i=1}^{\tilde{I}} p((\tilde{f}, \tilde{e})_i | (\tilde{f}, \tilde{e})_{i-n+1}^{i-1}). \end{aligned}$$

The Viterbi alignments for both translation directions are obtained using GIZA++ (Och and Ney, 2003), merged and then used to convert the bilingual sentence pairs into JTR sequences. A 7-gram JTR joint model (Guta et al., 2015), which is responsible for estimating the translation and reordering probabilities, is trained on those. It is estimated with interpolated modified Kneser-Ney smoothing (Chen and Goodman, 1998) using the KenLM toolkit (Heafield et al., 2013).

4.2.2 Language Models

The phrase-based translation system uses two language models (LM) that are estimated with the KenLM toolkit (Heafield et al., 2013) and integrated into the decoder as separate models in the log-linear combination: A 5-gram LM and a 7-gram word-class language model (wcLM). Both use interpolated modified Kneser-Ney smoothing. For the word-class LM, we train 200 word classes on the target side of the bilingual training data using an in-house tool (Botros et al., 2015) similar to `mkcls` (Och, 2000). We have not tuned the number of word classes, but simply used 200, as it has proved to work well in previous systems. With these class definitions, we apply the technique described in (Wuebker et al., 2013) to estimate the wcLM on the same data as the conventional LM.

Both models are trained on all monolingual corpora, except the commoncrawl corpus, and the target side of the bilingual data (Section 4.2), which sums up to 365.44M sentences and 7230.15M running words, respectively.

4.2.3 Log-Linear Features in Decoding

In addition to the JTR model and the language models, JTR conditional models for both directions (Peter et al., 2016) are included into the log-linear framework. They are computed offline on the phrase level. Moreover, the system incorporates phrase translation models estimated as relative frequencies for both directions.

Because the JTR models are trained on Viterbi aligned word-pairs, they are limited to the context provided by the aligned word pairs and sensitive to the quality of the word alignments. To overcome this issue, we incorporate IBM 1 lexical models for both directions. The models are trained on all available bilingual data and the synthetic data, see Section 3.

The heuristic features used by the decoder are an enhanced low frequency penalty (Chen et al.,

2011), a penalty for unaligned source words and a symmetric word-level distortion penalty. Thus, different phrasal segmentations have the same reordering costs if they are equal in their word alignments. An additional word bonus helps to control the length of the hypothesized translation by counteracting the language model, which prefers translations to be rather short.

The decoder also incorporates a gap distance penalty (Durrani et al., 2011). All parameter weights are optimized using MERT (Och, 2003) towards the BLEU metric.

An attention-based recurrent neural model is applied as an additional feature in rescoring 1000-best lists, see Section 4.2.4.

4.2.4 Attention-based Recurrent Neural Network in Re-Ranking

An attention-based recurrent neural network similar to those in Subsection 4.1 is used within the log-linear framework for rescoring 1000-best lists generated by the phrasal JTR decoder. The model is trained on 6.96M sentences of the synthetic data.

The network uses the 30K most frequent words as source and target vocabulary, respectively. The decoder and encoder word embeddings are of size 500, the encoder uses a bidirectional LSTM layer with 1K units (Hochreiter and Schmidhuber, 1997b) to encode the source side. An LSTM layer with 1K units is used by the decoder.

Training is performed for up to 300K iterations with a batch size of 50 and Adam (Kingma and Ba, 2014) is used as the optimization algorithm. The parameters of the best four networks on newstest2015 with regards to BLEU score are averaged to produce the final model used in reranking.

4.2.5 Alignment-based Recurrent Neural Network in Re-Ranking

Besides the attention-based model, we apply recurrent alignment-based neural networks in 1000-best rescoring. These networks are similar to the ones used in rescoring in (Alkhouli et al., 2016).

We use a bidirectional alignment model that has a bidirectional encoder (2 LSTM layers), a unidirectional target encoder (1 LSTM layer), and an additional decoder LSTM layer. The model pairs each target state computed at target position $i-1$ with its aligned bidirectional source state. The alignment information is obtain using GIZA++ in training, and from the 1000-best lists during rescoring. The paired states are fed into the decoder layer. The model predicts the discrete jump from the previous to the current source position. The model is described in (Alkhouli and Ney, 2017).

We also use a bidirectional lexical model to score word translation. It uses an architecture similar to that of the alignment model, with the exception that pairing is done using the source states aligned to the target position i instead of $i-1$. We also add weighted residual connections connecting the target states and the decoder states in the lexical model. We train two variants of this model, one including the target state, and one dropping it completely.

All models use four 200-node LSTM layers with the exception of the lexical model that includes the target state, which uses 350 nodes per layer. We use a class-factored output layer of 2000 classes, where 1000 classes are dedicated to the most frequent words, while the remaining 1000 classes are shared. This enables handling large vocabularies. The target vocabulary is reduced to 269K words, while the source vocabulary is reduced to 317K words

4.3 System Combination

System combination is applied to produce consensus translations from multiple hypotheses obtained from different translation approaches. The consensus translations typically outperform the individual hypotheses in terms of translation quality. A system combination implementation developed at RWTH Aachen University (Freitag et al., 2014) is used to combine the outputs of the different engines.

The first step in system combination is the generation of confusion networks (CN) from I input translation hypotheses. We need pairwise alignments between the input hypotheses. The alignments are obtained by METEOR (Banerjee and Lavie, 2005). The hypotheses are then reordered to match a selected skeleton hypothesis regarding the order of words. We generate I different CNs, each having one of the input systems as the skeleton hypothesis. The final lattice is the union of all I-many generated CNs.

The decoding of a confusion network consists of finding the shortest path in the network. Each arc is assigned a score of a linear model combination of M different models, which includes a

Individual Systems	newstest2015				newstest2016				newstest2017			
	BLEU	TER	cTER	BEER	BLEU	TER	cTER	BEER	BLEU	TER	cTER	BEER
Phrasal JTR + LM	29.7	52.5	50.5	61.3	33.9	48.0	46.5	64.2	29.4	53.1	51.6	61.2
+ wcLM	30.3	51.9	50.4	61.5	34.2	47.3	46.3	64.4	30.0	52.2	51.2	61.4
+ attention NMT	31.3	51.1	49.3	61.9	35.3	46.5	45.3	64.6	31.0	51.3	50.4	61.7
+ attention NMT	31.0	51.2	49.5	61.8	35.0	46.7	45.3	64.7	30.6	51.7	50.5	61.7
+ alignment NMT (x3)	30.9	51.0	49.6	61.8	35.3	46.5	45.6	64.8	30.7	51.6	50.7	61.7
+ attention NMT	31.3	50.9	49.3	61.9	35.3	46.4	45.3	64.8	30.9	51.2	50.2	61.8
NMT ensemble	32.0	48.9	47.5	62.3	38.8	42.7	42.5	66.2	33.1	48.3	47.7	63.0
System Combination	31.9	49.4	48.0	62.1	38.0	43.5	43.1	65.8	32.7	48.6	48.1	62.7

Table 3: Results of the individual systems for the German→English task. The system combination contains the system in line 3, 6, and 7.

word penalty, a 3-gram LM 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 the system outputs 1 if the decoded word origins from that system and 0 otherwise.

The model weights for the system combination are trained with MERT.

5 Experimental Evaluation

We have mainly focused on building a strong German→English system and run most experiments on this task. We used newstest2015 as the development set.

After switching the preprocessing as described in Section 2, we have added the word fertility, which improves the baseline system by about 0.8 BLEU on newstest2016 as shown in Table 2. Adding the synthetic data as described in Section 3 gives a gain of 3.8 BLEU on newstest2016. Changing the number of layers in the decoder from one to two improves the performance by additional 0.8 BLEU. Filtering the rapid data corpus by scoring all bilingual sentences with an NMT system trained on all parallel data and removing the sentences with the worst scores improves the system on newstest2016 by 0.4 BLEU, but yield only in a small improvement on newstest2015. Surprisingly, it even decreases the performance on newstest2017, as observed at a later point in time. Switching from merging the 4 best networks in a training run to continuing the training with an annealing scheme for learning rate reduction for SGD, as described in (Bahar et al., 2017), has barely changed the performance on newstest2016. Nevertheless, we have decided to keep on using it, since it slightly helped on newstest2015.

We have used this, without the word fertility, as a base setup to train multiple systems with slightly different settings for an ensemble. In the first setting we use all LSTM states of the first decoder layer as input for the second decoder layer. This actually hurts the performance. Adding the word fertility or the alignment feedback as additional information does not have a large impact. Note, that the word fertility helpes when it is added to the baseline system - we are not sure why the effect disappears. Combining systems in one ensemble improves the system again by 1.1 BLEU on newstest2016.

We also combined the NMT system with the strongest phrasal JTR system and a few other combinations as well, but none of them has been able to improve over the NMT ensemble (Table 3). We therefore used the NMT system as our final submission. In the table, we can see that using three alignment-based models is comparable to using a single attention-based model. Note, however, that these models have relatively small LSTM layers of 200 and 350 nodes per layer. Meanwhile, the attention model uses 1000-node LSTM layers. When added on top of the alignment-based mix, the attention model only improves the mix slightly.

For the English→German system we have simply used the three best working NMT systems from the German→English setup and combined them in an ensemble. The word fertility and alignment feedback extensions also did not improve the performance, but the ensemble increased the overall performance by 1 BLEU on newstest2016. Due to computation time limitations, we did not succeed in building a phrasal JTR system on time.

6 Conclusion

The RWTH Aachen University has participated with a neural machine translation ensemble for the

Individual Systems	newstest2015				newstest2016				newstest2017			
	BLEU	TER	CTER	BEER	BLEU	TER	CTER	BEER	BLEU	TER	CTER	BEER
NMT	26.7	54.7	50.9	60.0	31.8	48.4	46.6	63.6	25.4	56.2	52.8	59.5
+ fertility	26.8	54.8	50.5	60.1	31.5	48.6	46.7	63.4	25.3	56.2	52.9	59.5
+ alignment feedback	26.3	55.5	51.5	59.7	31.3	48.8	47.0	63.3	25.1	56.7	53.1	59.3
Ensemble	27.4	54.1	50.2	60.4	32.8	47.4	45.7	64.1	26.0	55.5	51.9	59.9

Table 4: Results of the individual systems for the English→German task.

German→English and English→German WMT 2017 evaluation campaign. All networks are trained using all given parallel data, back-translated synthetic data, two LSTM layers in the decoder. The rapid corpus has been filtered to remove the most unlikely sentences. Adam followed by annealing scheme of learning rate reduction is used for optimization. Four networks are combined for the German→English ensemble and three for the English→German ensemble. In addition, we have submitted a phrasal JTR system, which has come close to the performance of a single neural machine translation network for newstest2017. Using system combination has not improved the performance of the best neural ensemble.

Acknowledgements

The work reported in this paper results from two projects, SEQCLAS and QT21. SEQCLAS has received funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme under grant agreement n° 694537. QT21 has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement n° 645452. The work reflects only the authors' views and neither the European Commission nor the European Research Council Executive Agency are responsible for any use that may be made of the information it contains. Tamer Alkhouli was partly funded by the 2016 Google PhD Fellowship for North America, Europe and the Middle East.

References

Tamer Alkhouli, Gabriel Bretschner, Jan-Thorsten Peter, Mohammed Hethnawi, Andreas Guta, and Hermann Ney. 2016. Alignment-based neural machine translation. In *Proceedings of the First Conference on Machine Translation*. Association for Computational Linguistics, Berlin, Germany, pages 54–65. http://www.aclweb.org/anthology/W16-2206.

Tamer Alkhouli and Hermann Ney. 2017. Biasing attention-based recurrent neural networks using external alignment information. In *Proceedings of the Second Conference on Machine Translation*. Association for Computational Linguistics, Copenhagen, Denmark.

Parnia Bahar, Tamer Alkhouli, Jan-Thorsten Peter, Christopher Brix, and Hermann Ney. 2017. Empirical investigation of optimization algorithms in neural machine translation. In *Conference of the European Association for Machine Translation*. Prague, Czech Republic, pages 13–26.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2015. Neural Machine Translation by Jointly Learning to Align and Translate. In *Proceedings of the International Conference on Learning Representations (ICLR)*. San Diego.

Satanjeev Banerjee and Alon Lavie. 2005. 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, pages 65–72.

Frédéric Bastien, Pascal Lamblin, Razvan Pascanu, James Bergstra, Ian J. Goodfellow, Arnaud Bergeron, Nicolas Bouchard, and Yoshua Bengio. 2012. Theano: new features and speed improvements. Deep Learning and Unsupervised Feature Learning NIPS 2012 Workshop.

James Bergstra, Olivier Breuleux, Frédéric Bastien, Pascal Lamblin, Razvan Pascanu, Guillaume Desjardins, Joseph Turian, David Warde-Farley, and Yoshua Bengio. 2010. Theano: a CPU and GPU math expression compiler. In *Proceedings of the Python for Scientific Computing Conference (SciPy)*. Oral Presentation.

Nicola Bertoldi and Marcello Federico. 2009. Domain adaptation for statistical machine translation with monolingual resources. In *Proceedings of the Fourth Workshop on Statistical Machine Translation*. Association for Computational Linguistics, Stroudsburg, PA, USA, StatMT '09, pages 182–189. http://dl.acm.org/citation.cfm?id=1626431.1626468.

Rami Botros, Kazuki Irie, Martin Sundermeyer, and Hermann Ney. 2015. On efficient training of word classes and their application to recurrent neural network language models. In *Interspeech*. Dresden, Germany, pages 1443–1447.

Boxing Chen, Roland Kuhn, George Foster, and Howard Johnson. 2011. Unpacking and Transforming Feature Functions: New Ways to Smooth Phrase Tables. In *MT Summit XIII*. Xiamen, China, pages 269–275.

Stanley F. Chen and Joshuo Goodman. 1998. An Empirical Study of Smoothing Techniques for Language Modeling. Technical Report TR-10-98, Computer Science Group, Harvard University, Cambridge, MA.

Trevor Cohn, Cong Duy Vu Hoang, Ekaterina Vymolova, Kaisheng Yao, Chris Dyer, and Gholamreza Haffari. 2016. Incorporating structural alignment biases into an attentional neural translation model. *CoRR* abs/1601.01085. http://arxiv.org/abs/1601.01085.

Nadir Durrani, Helmut Schmid, and Alexander Fraser. 2011. A joint sequence translation model with integrated reordering. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*. Portland, Oregon, USA, pages 1045–1054. http://www.aclweb.org/anthology/P11-1105.

Markus Freitag, Matthias Huck, and Hermann Ney. 2014. 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, pages 29–32.

Andreas Guta, Tamer Alkhouli, Jan-Thorsten Peter, Joern Wuebker, and Hermann Ney. 2015. 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*. Association for Computational Linguistics, Lisbon, Portugal.

Kenneth Heafield, Ivan Pouzyrevsky, Jonathan H. Clark, and Philipp Koehn. 2013. Scalable modified Kneser-Ney language model estimation. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics*. Sofia, Bulgaria, pages 690–696.

Sepp Hochreiter and Jürgen Schmidhuber. 1997a. Long short-term memory. *Neural Comput.* 9(8):1735–1780. https://doi.org/10.1162/neco.1997.9.8.1735.

Sepp Hochreiter and Jürgen Schmidhuber. 1997b. Long short-term memory. *Neural computation* 9(8):1735–1780.

Marcin Junczys-Dowmunt, Tomasz Dwojak, and Rico Sennrich. 2016. The AMU-UEDIN submission to the WMT16 news translation task: Attention-based NMT models as feature functions in phrase-based SMT. In *Proceedings of the First Conference on Machine Translation, WMT 2016, colocated with ACL 2016, August 11-12, Berlin, Germany*. pages 319–325. http://aclweb.org/anthology/W/W16/W16-2316.pdf.

Diederik P. Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *CoRR* abs/1412.6980.

Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, Chris Dyer, Ondřej Bojar, Alexandra Constantine, and Evan Herbst. 2007. Moses: Open Source Toolkit for Statistical Machine Translation. Prague, Czech Republic, pages 177–180.

Franz J. Och. 2000. mkcls: Training of word classes for language modeling.

Franz Josef Och. 2003. Minimum Error Rate Training in Statistical Machine Translation. In *Proc. of the 41th Annual Meeting of the Association for Computational Linguistics (ACL)*. Sapporo, Japan, pages 160–167.

Franz Josef Och and Hermann Ney. 2003. A Systematic Comparison of Various Statistical Alignment Models. *Computational Linguistics* 29(1):19–51.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a Method for Automatic Evaluation of Machine Translation. In *Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics*. Philadelphia, Pennsylvania, USA, pages 311–318.

Jan-Thorsten Peter, Andreas Guta, Nick Rossenbach, Miguel Graa, and Hermann Ney. 2016. The rwth aachen machine translation system for iwslt 2016. In *International Workshop on Spoken Language Translation*. Seattle, USA.

Jan-Thorsten Peter, Farzad Toutounchi, Joern Wuebker, and Hermann Ney. 2015. The rwth aachen german-english machine translation system for wmt 2015. In *EMNLP 2015 Tenth Workshop on Statistical Machine Translation*. Lisbon, Portugal, page 158163.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016a. Edinburgh neural machine translation systems for wmt 16. In *Proceedings of the First Conference on Machine Translation*. Berlin, Germany, volume 2: Shared Task Papers, pages 368–373.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016b. Improving neural machine translation models with monolingual data .

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016c. Neural machine translation of rare words with subword units pages 1715–1725.

Matthew Snover, Bonnie Dorr, Richard Schwartz, Linnea Micciulla, and John Makhoul. 2006. 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, pages 223–231.

Miloš Stanojević and Khalil Sima'an. 2014. Fitting sentence level translation evaluation with many dense features. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Association for Computational Linguistics, Doha, Qatar, pages 202–206. http://www.aclweb.org/anthology/D14-1025.

Zhaopeng Tu, Zhengdong Lu, Yang Liu, Xiaohua Liu, and Hang Li. 2016. Coverage-based neural machine translation. *CoRR* abs/1601.04811. http://arxiv.org/abs/1601.04811.

Bart van Merriënboer, Dzmitry Bahdanau, Vincent Dumoulin, Dmitriy Serdyuk, David Warde-Farley, Jan Chorowski, and Yoshua Bengio. 2015. Blocks and fuel: Frameworks for deep learning. *CoRR* abs/1506.00619. http://arxiv.org/abs/1506.00619.

Weiyue Wang, Jan-Thorsten Peter, Hendrik Rosendahl, and Hermann Ney. 2016. Character: Translation edit rate on character level. In *ACL 2016 First Conference on Machine Translation*. Berlin, Germany, pages 505–510.

Joern Wuebker, Stephan Peitz, Felix Rietig, and Hermann Ney. 2013. Improving statistical machine translation with word class models. In *Conference on Empirical Methods in Natural Language Processing*. Seattle, WA, USA, pages 1377–1381.

Richard Zens and Hermann Ney. 2008. Improvements in Dynamic Programming Beam Search for Phrase-based Statistical Machine Translation. In *International Workshop on Spoken Language Translation*. Honolulu, Hawaii, pages 195–205.