The RWTH Aachen Machine Translation Systems for IWSLT 2017

Parnia Bahar, Jan Rosendahl*, Nick Rossenbach and Hermann Ney*

Human Language Technology and Pattern Recognition Group
Computer Science Department
RWTH Aachen University
Aachen, Germany

<surname>@cs.rwth-aachen.de

*Authors contributed equally

Abstract

This work describes the Neural Machine Translation (NMT) system of the RWTH Aachen University developed for the English↔German tracks of the evaluation campaign of the International Workshop on Spoken Language Translation (IWSLT) 2017. We use NMT systems which are augmented by state-of-the-art extensions. Furthermore, we experiment with techniques that include data filtering, a larger vocabulary, two extensions to the attention mechanism and domain adaptation. Using these methods, we can show considerable improvements over the respective baseline systems and our IWSLT 2016 submission.

1. Introduction

We describe the Neural Machine Translation (NMT) system of the RWTH Aachen University developed for the evaluation campaign of International Workshop on Spoken Language Translation (IWSLT) 2017. We have participated in the unofficial bilingual Machine Translation (MT) track for the German→English and English→German language pairs. The in-house NMT system incorporates various state-of-the-art extensions.

For the IWSLT 2016 evaluation campaign, RWTH Aachen utilized different translation systems [1] including a state-of-the-art phrase-based system, a neural machine translation system and the joint translation and reordering (JTR) model [2]. Furthermore, last year's system applied feed-forward and recurrent neural language and translation models for reranking. The attention-based approach had been used for reranking the n-best lists for both the phrase-based and the hierarchical setups. On top of these systems, a system combination enhances the translation quality by combining individually trained systems. For the IWSLT 2017 evaluation campaign, we developed the systems only based on the NMT approach as it has shown the most promising results among all.

This paper is organized as follows. In Section 2, we briefly address our preprocessing which differs from our previous submissions [3, 1]. Section 3 describes the details of the NMT systems, the baseline, our optimization techniques as well as two extensions to the attention mechanism. Our experiments for each track are summarized in Section 4.

2. Preprocessing

2.1. Preprocessing

Recent studies [4] showed that attention-based neural network systems do not benefit from several established preprocessing features such as compound splitting and POS-based word reordering. Therefore, we decided to employ a simpler version of preprocessing which uses only tokenization, frequent casing, and simple categories. In this approach, numbers are not mapped to a specific category-token but are treated like regular words instead.

All words and numbers are split into subword units using byte-per-encoding (BPE)[1] introduced by [5]. We use 90k BPE merging operations trained jointly on the concatenated source and target training data. In the preprocessing, we do not distinguish if a language is seen as a source or target language.

2.2. Data Filtering

In order to remove incorrectly aligned sentence pairs, we drop all training samples for which the length of the source sentence exceeds the length of the target sentence by more than about 70%. We applied this method for both translation directions. In the following we describe the effects for the English→German task. The length comparison is executed on the word level and results in the total removal of 1.7M sentences, i.e. 8% of the total training data. The majority of removed sentence pairs are part of the Common Crawl (300k sentences i.e. 14% of Common Crawl) and the OpenSubtitles corpora (1000k sentences i.e. 8% of OpenSubtitles). The removal rates for the individual corpora can be found in Table 1.

[1] https://github.com/rsennrich/subword-nmt

Table 1: Effect of filtering on the individual training corpora on the example of English→German.

	Common Crawl	Europarl	UN	News Comment	OpenSub	QED	TED	Wiki	**Total**
# sentences	2,399,123	1,920,209	162,981	216,190	13,430,645	72,747	206,112	2,459,662	**20,867,669**
# removed	336,248	38,032	3,323	3,547	1,056,628	4,343	2,560	218,860	**1,663,541**
% removed	14.02%	1.98%	2.04%	1.64%	7.87%	5.97%	1.24%	8.90%	**7.97 %**

3. Neural Machine Translation System

The best performing system provided by RWTH Aachen is an attention-based recurrent neural network similar to [6]. Provided with a source f_1^J and a target sequence e_1^I, NMT models the conditional probability of the target given the source. The model itself consists of an encoder which produces a continuous representation of the input sequence f_1^J, an attention mechanism which allows the system to focus on certain words during the translation and a decoder which returns a probability distribution over all possible target tokens for every time step.

3.1. Baseline System

We use the attention-based NMT system as our baseline. In our setup, all words are projected into a 620-dimensional embedding space both on the source and on the target side. The bidirectional encoder and the unidirectional decoder consist of LSTM nodes [7] with peephole connections using 1000 cells. The output layer of the networks is a two-layer maxout [8] followed by a softmax operation that creates a probability distribution over the target vocabulary. We use the additive attention with tanh activation function as proposed in [6] followed by the softmax to compute the attention weights.

3.2. Stacked Layers

In this architecture, we experiment with two stacked LSTM layers in both encoder and decoder to build a deeper model. We connect all internal states of the first LSTM layer to the second. This approach is applied both in the bidirectional encoder and the unidirectional decoder.

3.3. Optimization

Since the learning trajectory considerably depends on the optimization technique, the optimizer plays an important role in fast convergence, training stability and reliable performance. It is desired to have a fast convergence to a zone in which a good local minimum is located. After that, the algorithm shrinks the learning rate to get a finer search pattern and converge to a suitable model within the located area.

As proposed in [9], we start the training using Adam [10] with a learning rate of 0.001 up to 600k iterations. Afterwards the learning rate of the Adam optimizer is scaled down by the factor of 0.75 every 20k iterations. In the following, we refer to this approach by annealing Adam.

3.4. Fertility Feedback

One of the problems arising from the attention-based sequence-to-sequence model, which is used as our baseline, is that there is no explicit alignment or coverage information. The attention weights are included in the context vector and there is no guarantee that the network can extract this information in the next attention computation. One of the proposed solutions [11] is to feed back the sum of the alignments over the past decoder steps. This information is added to the computation of the attention energies for each source position. Hence, in each decoder step this sum indicates how much attention has been given to the source position j up to step i. The feedback term $\hat{\beta}_{i,j}$ is expressed as:

$$\hat{\beta}_{i,j} = \sum_{k=1}^{i-1} \alpha_{k,j} \tag{1}$$

One might simply use $\hat{\beta}_{i,j}$ as an additional information in order to compute the attention energies. Instead, we use a fertility parameter that determines how many target words should be generated by a single source word. The concept of fertility has been introduced in IBM Model 3 and can be integrated into neural networks [11, 12].

Let's assume a single word should be translated twice, then $\hat{\beta}_{i,j}$ can be divided by a factor of 2. This normalizes the sum presented in Equation 1, such that the network can use the information whether the current word is over- or under-translated. Therefore, $\beta_{i,j}$ is defined as:

$$\beta_{i,j} = \frac{1}{\phi_j} \sum_{k=1}^{i-1} \alpha_{i,j} \tag{2}$$

where ϕ_j refers to the fertility of f_j. This term depends on the encoder states, because it can vary if the word is used in a different context. Like [11] in our model ϕ_j is defined as:

$$\phi_j = N \cdot \sigma(\upsilon_\phi^\top \cdot h_j) \tag{3}$$

where N specifies the maximum value for the fertility which is set to 2 in our experiments. This value is included in the calculation of the attention energies $e_{i,j}$:

$$e_{i,j} = \upsilon^\top \tanh\left(W s_{i-1} + U h_j + V \beta_{i,j}\right) \tag{4}$$

where h_j and s_i denote the output of the encoder and the decoder state respectively. W, U, V and υ are the weight matrices.

3.5. Convolutional Feedback

In the standard attention-based model, there is no dependency on the source position while computing the attention weights. Several authors argue [13, 14] that this independence assumption does not hold for monotonous alignments as can be found in speech recognition. Although the alignments in machine translation are not monotonous in general, we still encounter many cases of local monotonicity in many languages. Convolutional attention feedback tries to encounter such problems by putting an explicit focus on the source positions around j when generating the j-th target word. Formally, it computes feature vectors γ_i by applying a one-dimensional convolutional operation over the attention weights from the last decoder step:

$$\gamma_i = G * \alpha_{i-1} \tag{5}$$

where $G \in \mathbb{R}^{N \times 2k+1}$. This leads to N vectors, one for each filter that has been applied. Every filter moves over a window of size $2k+1$ that is centered at position j, i.e.:

$$\gamma_{i,j} = \sum_{l=j-k}^{j+k} G_{n,j-l} \cdot \alpha_{i,l} \qquad \text{for all } n = 1, \cdots, N. \tag{6}$$

The result of this is used as a feedback term to compute the attention weights in the current decoder step:

$$e_{i,j} = v^\top \tanh\left(Ws_{i-1} + Uh_j + V\gamma_{i,j}\right). \tag{7}$$

We use 5 filters with a window of size 5 in our experiments that include convolutional feedback. Again we use h_j respectively s_i to denote the output of the encoder and the internal state of the decoder.

4. Experimental Evaluation

For the evaluation, we carry out experiments on two translation tasks: German→English and English→German. The translation systems are built using our in-house implementation of the attention-based NMT approach which relies on the Blocks[2] framework [15] and Theano[3] [16].

All systems are trained on the filtered bilingual data as described in Section 2.2 and no monolingual data. In order to adapt our system to the domain of TED Talks, we add the TED corpus eleven times and the QED corpus six times to our training data. This results in a training set of 21.6M parallel sentence pairs.

Before training, we shuffle the training samples once and use mini-batches of 50 sentence pairs while sentences longer than 65 subwords are dropped. The processing of one mini-batch is called an iteration. The networks are trained for up to 600k iterations and equipped with the various features presented in Section 3. We evaluate the models every 10k iterations.

Throughout our experiments, we observe that employing the Adam annealing scheme consistently gives us strong improvements of at least 1.5% BLEU over the pure Adam optimizer. Similar gains can be achieved by averaging the weights among the four best models of a single training run as described in the beginning of [17]. Both methods are applied to improve upon a weak Adam endpoint. Hence, we always pick the option that leads to a better average BLEU score. The results of the other method are omitted in this paper for the sake of brevity.

We try to fine-tune the models on the indomain data which consists of the TED corpus to which `TED.tst2011`, `TED.tst2012` and `TED.tst2013` sets were added.

Decoding is performed using beam search with a beam size of 12 and the scores are normalized w.r.t the length of the hypotheses.

We use `TED.dev2010` consisting of 888 sentences as our validation set and evaluate our models on `TED.tst2010`, `TED.tst2014` and `TED.tst2015` as unseen test sets. The systems are evaluated using case-sensitive BLEU [18] computed by `mteval-v13a` [4], TER [19] computed by `tercom` [5] and CharacterTER [20] which we abbreviate to CTER[6].

To avoid the out of vocabulary problem, we use the joint BPE [5] to convert sentences into the sequences of subwords on both the source and the target side. In both tasks, the number of joint-BPE merging operations is 90k.

4.1. German→English

Based on the work done in [4], we equip the German→English baseline with two layers of stacked LSTMs in both the encoder and the decoder which is referred to as `multilayer enc-dec baseline`. The total number of parameters for this setup is about 220M. All networks are trained with 30% of dropout for better regularization. The results are depicted in Table 2, Row 1. After training the network and reaching convergence, we apply annealing Adam as mentioned in Section 3.3 for additional 300k iterations (Row 2 in Table 2). As shown, this strategy results in improvements up to 2.4% BLEU score, 1.4% TER and 1.4% CTER averaged over the four test sets.

Using additional information from previous attention states by employing fertility feedback, we gain 0.5% BLEU and 0.1% TER on average. The results in Row 3 of Table 2 have been obtained by applying annealing Adam. On top of this model, we fine-tune the system. Here, we pick the best model and retrain it using the indomain TED data discussed before for around 20 epochs. Surprisingly, fine tuning does not help and even hurts slightly in terms of TER. One of the reasons is that we have already weighted our indomain data in the training data such that any further fine tuning does not affect the learning trajectory. In the other words, the model

[2] https://github.com/mila-udem/blocks-examples
[3] http://deeplearning.net/software/theano/
[4] ftp://jaguar.ncsl.nist.gov/mt/resources/mteval-v13a.pl
[5] http://www.cs.umd.edu/ snover/tercom/
[6] https://github.com/rwth-i6/CharacTER

Table 2: Results measured in BLEU [%], TER [%] and CTER [%] for the individual systems for the German→English MT task.

#	System	TED.dev2010			TED.tst2010			TED.tst2014			TED.tst2015		
		BLEU	TER	CTER	BLEU	TER	CTER	BLEU	TER	CTER	BLEU	TER	CTER
1	multilayer enc-dec baseline	34.3	44.9	46.1	33.0	46.0	48.1	31.3	49.0	51.6	31.1	48.2	50.6
2	+ annealing Adam	36.4	43.2	45.2	35.0	44.3	46.3	33.8	46.2	49.3	34.0	46.0	48.3
3	+ fertility feedback	37.0	42.4	44.8	35.6	43.7	45.3	33.9	46.2	49.0	34.6	45.4	48.4
4	+ fine tuned	36.2	43.2	44.9	35.6	44.0	45.2	33.9	46.6	48.3	34.5	45.7	48.6
5	+ convolutional feedback (averaged4)	36.2	43.0	45.4	34.9	44.1	46.5	33.1	46.4	49.2	33.1	45.8	49.0
6	ensemble 2, 3, 5	38.3	41.2	43.4	37.3	42.3	44.3	35.5	44.9	47.8	35.5	44.5	47.6

is smoothly adapted to the TED domain from the first iterations. We also apply convolutional feedback as explained in Section 3.5, and average the best four models of a single training run (see Row 5). As it can be seen, convolutional feedback is slightly better in terms of TER and hurts in terms of BLEU compared to Row 2.

Finally, we build an ensemble [21] of different architectures including two multilayer enc-dec baseline, fertility feedback and convolutional feedback models. Ensemble improves the translation performance compared to the best system (Row 3) by 1.4% in terms of BLEU, 0.6% TER and 0.4% CTER on average.

4.2. English→German

For the English→German task we start with a simple baseline described in Section 3.1 which employs a single LSTM-layer for both the bidirectional encoder and the decoder. The model is trained using the Adam algorithm for 600k iterations and by default, no dropout is applied.

On top of this baseline, we add various feature combinations. Results are shown in Table 3. Adding dropout to the baseline system yields an average improvement of 0.7% BLEU. Based on this, we continue the training with the annealing Adam for 300k iterations which gives us an improvement of 2.5% BLEU.

Furthermore, we train a series of models that utilize the fertility feedback presented in Section 3.4. Adding this feature on top of the baseline system yields an improvement of 0.3% BLEU (Table 3, Row 4). Adding a second LSTM-layer to both the encoder and the decoder leads to an average gain of 0.2% BLEU and 1.1% TER.

Again, we observe that it is important to keep on training for 300k iterations with a small learning rate as this boosts our performance by 2.3% BLEU (Table 3, Row 7). Usually, the models extracted from a training run are among the last models saved during the 600k iterations. Therefore, the effect of the annealing Adam scheme can be attributed to an insufficiently small learning rate or a model that is not fully converged. However, it hurts the performance of the model if we further continue training on the regular training data.

We fine-tune the models either on the indomain data or an expanded version which contains the QED corpus as well. Both approaches led to almost no change w.r.t BLEU and TER. As in the case of the German→English system, we conclude that due to the weighting of the TED data, additional domain adaptation is of little use. However the models that are fine-tuned on the TED corpus perform a little bit stronger in the final ensemble which is why we decide to keep them.

In total, we combine fertility feedback, multi-layered encoder and decoder as well as dropout with an annealing version of Adam to get an improvement of 3.3% BLEU (Table 3, Row 9). Surprisingly, by averaging the four best fertility feedback models (Table 3, Row 5), we obtain a smaller model that has been trained for a much shorter period of time but performs only 0.3% BLEU worse than to the fine-tuned one on average.

Combining several of the systems in one ensemble led to an average improvement of 1.5% BLEU and 1.4% TER over our single best system.

4.3. Final Results

Compared to last year's submission, we have completely moved towards pure neural MT systems. Although last year's system contains a phrase-based system in combination with the JTR model [2], neural language and translation models as well as NMT systems, the results are improved by 2.3% BLEU and 1.8% TER for the `TED.tst2010` set and by 1.3% BLEU and 1.6% TER on the `TED.tst2014` set as shown in Table 4. Furthermore, the pure NMT system for 2017 submission shows a huge improvement compared to the 2015 submission in which the NMT model had only been used in the reranking of the n-best lists for both phrase-based and hierarchical setups.

The performance on the `TED.tst2016` and `TED.tst2017` test sets is shown in Table 5. We evaluate our hypothesis via the IWSLT 2017 evaluation server.

5. Conclusion

The RWTH Aachen has participated in two bilingual MT tracks for the German→English and English→German IWSLT 2017 evaluation campaign. The 2017 submission includes neural models only opposed to last year's system included the NMT system and the phrase-based system. The

Table 3: Results measured in BLEU [%], TER [%] and CTER [%] for the individual systems for the English→German MT task.

#	System	TED.dev2010			TED.tst2010			TED.tst2014			TED.tst2015		
		BLEU	TER	CTER	BLEU	TER	CTER	BLEU	TER	CTER	BLEU	TER	CTER
1	baseline	26.0	55.3	50.6	26.4	54.3	51.1	24.6	57.1	53.7	27.2	55.2	51.1
2	+ dropout	26.7	53.7	50.5	27.4	53.0	50.8	25.3	56.4	53.9	27.4	55.1	51.3
3	+ annealing Adam	28.6	52.1	47.1	30.2	51.0	48.5	27.9	53.9	50.7	30.2	53.0	48.4
4	+ fertility feedback	26.4	54.3	50.6	26.7	54.0	51.5	25.3	56.7	53.8	27.0	55.5	50.9
5	+ average4	28.9	50.9	47.0	29.9	50.9	47.7	27.2	54.3	50.5	29.9	52.4	47.9
6	+ multilayer enc-dec	26.5	53.5	50.0	27.1	53.5	51.3	25.2	56.2	53.5	27.3	54.5	50.7
7	+ annealing Adam	28.8	51.1	46.8	29.6	51.2	48.0	27.6	54.0	50.0	29.9	52.6	48.3
8	+ fine tuned	28.4	51.4	47.0	29.8	51.2	47.6	27.5	54.3	49.8	29.9	52.7	47.4
9	+ dropout	28.9	51.4	47.3	30.1	50.8	47.4	27.6	54.1	50.3	30.5	52.3	46.8
10	ensemble 3, 5, 8, 9	30.3	49.9	45.2	31.8	49.5	45.9	29.2	52.8	48.8	31.5	51.1	45.9

Table 4: Comparison to last years' German→English MT task submissions. Results measured in BLEU [%], TER [%] and NIST.

System	TED.tst2010			TED.tst2014		
	BLEU	TER	CTER	BLEU	TER	CTER
2015-Submission [3]	31.9	47.6	45.5	-	-	-
2016-Submission [1]	35.0	44.1	42.7	34.2	46.5	46.9
2017-Submission	37.3	42.3	44.3	35.5	44.9	47.8

Table 5: Results measured in BLEU [%], TER [%] and NIST on `TED.tst2016` and `TED.tst2017`.

MT Task	TED.tst2016			TED.tst2017		
	BLEU	TER	NIST	BLEU	TER	NIST
De→En	35.38	44.48	7.8947	30.22	49.44	7.1608
En→De	28.09	55.23	6.5995	25.12	59.09	6.1239

baseline systems for the MT track utilize our state-of-the-art attention-based neural machine translation. We are able to further improve translation by applying a multilayer encoder and decoder and increasing the number of subword units. Using refinements of the attention mechanism to feedback more alignment information leads to better results. A significant gain is achieved by the annealing scheme based on Adam and the ensemble of different NMT systems.

In total, we achieve a performance of 35.5% BLEU and 44.5% TER on the `TED.tst2015` data set of the German→English task. Compared to our 2016 submission, this is an improvement by 1.3% BLEU and 1.6% TER. For the English→German task our state-of-the-art system produces a score of 31.5% BLEU and 51.1% TER on `TED.tst2015`.

6. Acknowledgements

The work reported in this paper has been funded by three projects, SEQCLAS, QT21 and DFG-Core-Tec. 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. It was partially supported by the Deutsche Forschungsgemeinschaft (DFG) under Contract No NE572/8-1. The work reflects only the authors' views and neither the European Commission nor the European Research Council Executive Agency nor the DFG are responsible for any use that may be made of the information it contains.

7. References

[1] J.-T. Peter, A. Guta, N. Rossenbach, M. Graça, and H. Ney, "The rwth aachen machine translation system for iwslt 2016," in *International Workshop on Spoken Language Translation. Seattle, WA, USA*, 2016.

[2] 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.

[3] J.-T. Peter, F. Toutounchi, S. Peitz, P. Bahar, A. Guta, and H. Ney, "The rwth aachen german to english mt system for iwslt 2015," in *International Workshop on Spoken Language Translation*, Da Nang, Vietnam, Dec. 2015, pp. 15–22.

[4] J.-T. Peter, A. Guta, T. Alkhouli, P. Bahar, J. Rosendahl, N. Rossenbach, M. Graça, and H. Ney, "The rwth

aachen university english-german and german-english machine translation system for wmt 2017," in *Proceedings of the Second Conference on Machine Translation*, 2017, pp. 358–365.

[5] R. Sennrich, B. Haddow, and A. Birch, "Neural machine translation of rare words with subword units," in *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, ACL 2016, August 7-12, 2016, Berlin, Germany, Volume 1: Long Papers*, 2016. [Online]. Available: http://aclweb.org/anthology/P/P16/P16-1162.pdf

[6] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," May 2015. [Online]. Available: http://arxiv.org/abs/1409.0473

[7] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, Nov. 1997. [Online]. Available: http://dx.doi.org/10.1162/neco.1997.9.8.1735

[8] I. J. Goodfellow, D. Warde-Farley, M. Mirza, A. C. Courville, and Y. Bengio, "Maxout networks," in *Proceedings of the 30th International Conference on Machine Learning, ICML 2013, Atlanta, GA, USA, 16-21 June 2013*, 2013, pp. 1319–1327. [Online]. Available: http://jmlr.org/proceedings/papers/v28/goodfellow13.html

[9] P. Bahar, T. Alkhouli, J.-T. Peter, C. J.-S. Brix, and H. Ney, "Empirical investigation of optimization algorithms in neural machine translation," *The Prague Bulletin of Mathematical Linguistics, The 20th Annual Conference of the European Association for Machine Translation*, vol. 108, no. 1, pp. 13–25, 2017.

[10] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *CoRR*, vol. abs/1412.6980, 2014. [Online]. Available: http://arxiv.org/abs/1412.6980

[11] Z. Tu, Z. Lu, Y. Liu, X. Liu, and H. Li, "Coverage-based neural machine translation," *CoRR*, vol. abs/1601.04811, 2016. [Online]. Available: http://arxiv.org/abs/1601.04811

[12] T. Cohn, C. D. V. Hoang, E. Vymolova, K. Yao, C. Dyer, and G. Haffari, "Incorporating structural alignment biases into an attentional neural translation model," *CoRR*, vol. abs/1601.01085, 2016. [Online]. Available: http://arxiv.org/abs/1601.01085

[13] J. Chorowski, D. Bahdanau, D. Serdyuk, K. Cho, and Y. Bengio, "Attention-based models for speech recognition," *CoRR*, vol. abs/1506.07503, 2015. [Online]. Available: http://arxiv.org/abs/1506.07503

[14] S. Feng, S. Liu, M. Li, and M. Zhou, "Implicit distortion and fertility models for attention-based encoder-decoder NMT model," *CoRR*, vol. abs/1601.03317, 2016. [Online]. Available: http://arxiv.org/abs/1601.03317

[15] B. van Merriënboer, D. Bahdanau, V. Dumoulin, D. Serdyuk, D. Warde-Farley, J. Chorowski, and Y. Bengio, "Blocks and fuel: Frameworks for deep learning," *CoRR*, vol. abs/1506.00619, 2015. [Online]. Available: http://arxiv.org/abs/1506.00619

[16] 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.

[17] M. Junczys-Dowmunt, T. Dwojak, and R. Sennrich, "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*, 2016, pp. 319–325. [Online]. Available: http://aclweb.org/anthology/W/W16/W16-2316.pdf

[18] 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, 2002, pp. 311–318.

[19] 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, 2006, pp. 223–231.

[20] W. Wang, J.-T. Peter, H. Rosendahl, and H. Ney, "Character: Translation edit rate on character level," in *ACL 2016 First Conference on Machine Translation*, Berlin, Germany, Aug. 2016.

[21] S. Jean, O. Firat, K. Cho, R. Memisevic, and Y. Bengio, "Montreal neural machine translation systems for wmt'15," in *Proceedings of the Tenth Workshop on Statistical Machine Translation, WMT@EMNLP 2015, 17-18 September 2015, Lisbon, Portugal*, 2015, pp. 134–140.