The University of Cambridge's Machine Translation Systems for WMT18

Felix Stahlberg[†] and **Adrià de Gispert**[‡†] and **Bill Byrne**[‡†]
[†]Department of Engineering, University of Cambridge, UK
[‡]SDL Research, Cambridge, UK
{fs439,ad465,wjb31}@cam.ac.uk

Abstract

The University of Cambridge submission to the WMT18 news translation task focuses on the combination of diverse models of translation. We compare recurrent, convolutional, and self-attention-based neural models on German-English, English-German, and Chinese-English. Our final system combines all neural models together with a phrase-based SMT system in an MBR-based scheme. We report small but consistent gains on top of strong Transformer ensembles.

1 Introduction

Encoder-decoder networks (Pollack, 1990; Chrisman, 1991; Forcada and Ñeco, 1997; Kalchbrenner and Blunsom, 2013) are the current prevailing architecture for neural machine translation (NMT). Various architectures have been used in the general framework of encoder and decoder networks such as recursive auto-encoders (Pollack, 1990; Socher et al., 2011; Li et al., 2013), (attentional) recurrent models (Sutskever et al., 2014; Bahdanau et al., 2015; Luong et al., 2015; Wu et al., 2016; Chen et al., 2018), convolutional models (Kalchbrenner and Blunsom, 2013; Kaiser et al., 2017; Gehring et al., 2017), and, most recently, purely (self-)attention-based models (Vaswani et al., 2017; Ahmed et al., 2017; Shaw et al., 2018). In the spirit of Chen et al. (2018) we devoted our WMT18 submission to exploring the three most commonly used architectures: recurrent, convolutional, and self-attention-based models like the Transformer (Vaswani et al., 2017). Our experiments suggest that self-attention is the superior architecture on the tested language pairs, but it can still benefit from model combination with the other two. We show that using large batch sizes is crucial to Transformer training, and that the delayed SGD updates technique (Saunders et al., 2018) is useful to increase the batch size on limited GPU hardware. Furthermore, we also report gains from MBR-based combination with a phrase-based SMT system. We found this particularly striking as the SMT baselines are often more than 10 BLEU points below our strongest neural models. Our final submission ranks second in terms of BLEU score in the WMT18 evaluation campaign on English-German and German-English, and outperforms all other systems on a variety of linguistic phenomena on German-English (Avramidis et al., 2018).

2 System Combination

Stahlberg et al. (2017a) combined SMT and NMT in a hybrid system with a minimum Bayes-risk (MBR) formulation which has been proven useful even for practical industry-level MT (Iglesias et al., 2018). Our system combination scheme is a generalization of this approach to more than two systems. Suppose we want to combine q models $\mathcal{M}_1, \ldots, \mathcal{M}_q$. We first divide the models into two groups by selecting a p with $1 \leq p \leq q$. We refer to scores from the first group $\mathcal{M}_1, \ldots, \mathcal{M}_p$ as *full posterior* scores and from the second group $\mathcal{M}_{p+1}, \ldots, \mathcal{M}_q$ as *MBR-based* scores. Full posterior models contribute to the combined score with their complete posterior of the full translation. In contrast, models in the second group only provide the evidence space for estimating the probability of n-grams occurring in the translation. Full-posterior models need to assign scores via the standard left-to-right factorization of neural sequence models:

$$\log P(y_1^T|\mathbf{x}, \mathcal{M}_i) = \sum_{t=1}^{T} \log P(y_t|y_1^{t-1}, \mathbf{x}, \mathcal{M}_i) \quad (1)$$

for a target sentence $\mathbf{y} = y_1^T$ of length T given a source sentence $\mathbf{x}$ for all $i \leq p$. For exam-

Proceedings of the Third Conference on Machine Translation (WMT), Volume 2: Shared Task Papers, pages 504–512

https://doi.org/10.18653/v1/W18-64054

ple, all left-to-right neural models in this work can be used as full posterior models, but the right-to-left models (Sec. 3) and SMT cannot. We combine full-posterior scores log-linearly, and bias the combined score $S(\mathbf{y}|\mathbf{x})$ towards low-risk hypotheses with respect to the MBR-based group as suggested by Stahlberg et al. (2017a, Eq. 4):[1]

$$S(\mathbf{y}|\mathbf{x}) = \sum_{t=1}^{T} \Bigg(\underbrace{\sum_{i=1}^{p} \lambda_i \log P(y_t|y_1^{t-1}, \mathbf{x}, \mathcal{M}_i)}_{\text{Full posterior}} + \underbrace{\sum_{j=p+1}^{q} \lambda_j \sum_{n=1}^{4} P(y_{t-n}^{t}|\mathbf{x}, \mathcal{M}_j)}_{\text{MBR-based } n\text{-gram scores}} \Bigg) \quad (2)$$

where $\lambda_1, \ldots, \lambda_q$ are interpolation weights. Eq. 2 also describes how to use beam search in this framework as hypotheses can be built up from left to right due to the outer sum over time steps. The MBR-based models contribute via the probability $P(y_{t-n}^{t}|\mathbf{x}, \mathcal{M}_j)$ of an n-gram y_{t-n}^{t} given the source sentence $\mathbf{x}$. Posteriors in this form are commonly used for MBR decoding in SMT (Kumar and Byrne, 2004; Tromble et al., 2008), and can be extracted efficiently from translation lattices using counting transducers (Blackwood et al., 2010). For our neural models we run beam search with beam size 15 and compute posteriors over the 15-best list. We smooth all n-gram posteriors as suggested by Stahlberg et al. (2017a).

Note that our generalization to more than two systems can still be seen as instance of the original scheme from Stahlberg et al. (2017a) by viewing the first group $\mathcal{M}_1, \ldots, \mathcal{M}_p$ as ensemble and the evidence space from the second group $\mathcal{M}_{p+1}, \ldots, \mathcal{M}_q$ as mixture model.

The performance of our system combinations depends on the correct calibration of the interpolation weights $\lambda_1, \ldots, \lambda_q$. We first tried to use n-best or lattice MERT (Och, 2003; Macherey et al., 2008) to find interpolation weights, but these techniques were not effective in our setting, possibly due to the lack of diversity and depth in n-best lists from standard beam search. Therefore, we tune on the first best translation using Powell's method (Powell, 1964) with a line search algorithm similar to golden-section search (Kiefer, 1953).

[1] Eq. 2 differs from Eq. 4 of Stahlberg et al. (2017a) in that we do not use a word penalty Θ_0 here, and we do not tune weights for different order n-grams separately ($\Theta_1, \ldots \Theta_4$). Both did not improve translation quality in our setting.

3 Right-to-left Translation Models

Standard NMT models generate the translation from left to right on the target side. Recent work has shown that incorporating models which generate the target sentence in reverse order (i.e. from right to left) can improve translation quality (Liu et al., 2016; Li et al., 2017; Sennrich et al., 2017; Hassan et al., 2018). Right-to-left models are often used to rescore n-best lists from left-to-right models. However, we could not find improvements from rescoring in our setting. Instead, we extract n-gram posteriors from the R2L model, reverse them, and use them for system combination as described in Sec. 2.

4 Experimental Setup

4.1 Data Selection

We ran language detection (Nakatani, 2010) and gentle length filtering based on the number of characters and words in a sentence on all available monolingual and parallel data in English, German, and Chinese. Due to the high level of noise in the ParaCrawl corpus and its large size compared to the rest of the English-German data we additionally filtered ParaCrawl more aggressively with the following rules:

- No words contain more than 40 characters.
- Sentences must not contain HTML tags.
- The minimum sentence length is 4 words.
- The character ratio between source and target must not exceed 1:3 or 3:1.
- Source and target sentences must be equal after stripping out non-numerical characters.
- Sentences must end with punctuation marks.

This additional filtering reduced the size of ParaCrawl from originally 36M sentences to 19M sentences after language detection, and to 11M sentences after applying the more aggressive rules.

For backtranslation (Sennrich et al., 2016a) we selected 20M sentences from News Crawl 2017. We used a single Transformer (Vaswani et al., 2017) model in Tensor2Tensor's (Vaswani et al., 2018) `transformer_base` configuration

Corpus	Over-sampling	#Sentences
Common Crawl	2x	4.43M
Europarl v7	2x	3.76M
News Commentary v13	2x	0.57M
Rapid 2016	2x	2.27M
ParaCrawl	1x	11.16M
Synthetic (news-2017)	1x	20.00M
Total		**42.19M**

Table 1: Training data sizes for English-German and German-English after filtering.

Corpus	Over-sampling	#Sentences
CWMT - CASIA2015	2x	2.08M
CWMT - CASICT2015	2x	3.95M
CWMT - Datum2017	2x	1.93M
CWMT - NEU2017	2x	3.95M
News Commentary v13	2x	0.49M
UN v1.0	1x	14.25M
Synthetic (news-2017)	1x	20.00M
Total		**46.66M**

Table 2: Training data sizes for Chinese-English after filtering.

for generating the synthetic source sentences. We over-sampled (Sennrich et al., 2017) WMT data by factor 2 except the ParaCrawl data and the UN data on Chinese-English to roughly match the size of the synthetic data. Tabs. 1 and 2 summarize the sizes of our final training corpora.

4.2 Preprocessing

We preprocess our English and German data with Moses tokenization, punctuation normalization, and truecasing. On Chinese we first used the WMT `tokenizeChinese.py`[2] script and separated segments of Chinese and Latin text from each other. Then, we removed white-space between Chinese characters and tokenized Chinese segments with Jieba[3] and the rest with `mteval-v13a.pl`. For our neural models we apply byte-pair encoding (Sennrich et al., 2016b, BPE) with 32K merge operations. We use joint BPE vocabularies on English-German and German-English and separate source/target encodings on Chinese-English.

4.3 Model Hyper-Parameters

We use 1024-dimensional embedding and output projection layers in all architectures. The embeddings are shared between encoder and decoder on English-German and German-English, but not on Chinese-English.

LSTM For our recurrent models we adapted the TensorFlow seq2seq tutorial code base (Luong et al., 2017) for use inside the Tensor2Tensor library (Vaswani et al., 2018).[4] We roughly followed the UEdin WMT17 submission (Sennrich et al., 2017) and stacked four 1024-dimensional LSTM layers with layer normalization (Ba et al., 2016) and residual connections in both the decoder and bidirectional encoder. We equipped the decoder network with Bahdanau-style (Bahdanau et al., 2015) attention (`normed_bahdanau`).

SliceNet The convolutional model of Kaiser et al. (2017) called SliceNet is implemented in Tensor2Tensor. We use the standard configuration `slicenet_1` of four hidden layers with layer normalization.

Transformer We compare two Transformer variants available in Tensor2Tensor: the original Transformer (Vaswani et al., 2017) (`transformer_big` setup) and the Transformer of Shaw et al. (2018) with relative positional embeddings (`transformer_relative_big` setup). Both use 16-head dot-product attention and six 1024-dimensional encoder and decoder layers.

The number of training parameters of our neural models is summarized in Tab. 3.

Architecture	en-de, de-en	zh-en
LSTM	114.2M	192.7M
SliceNet	27.5M	86.4M
Transformer	212.8M	291.4M
Relative Transformer	213.8M	292.5M

Table 3: Number of model parameters.

#Physical GPUs (g)	Delay factor (d)	#Effective GPUs (g'=gd)	Effective batch size (b'=bg')	BLEU
1	1	1	2,048	28.2
4	1	4	8,192	29.5
4	4	16	32,768	30.3
4	16	64	131,072	29.8

Table 4: Impact of the effective batch size on Transformer training on en-de news-test2017 after 3,276M training tokens, beam size 4.

[2] `http://www.statmt.org/wmt17/tokenizeChinese.py`

[3] `https://github.com/fxsjy/jieba`

[4] `https://github.com/fstahlberg/tensor2tensor-usr`

Architecture	#Effective GPUs	Batch size	#SGD updates	#Training tokens
LSTM	8	4,096	45K	1,475M
SliceNet	4	2,048	800K	6,554M
R2L Transformer	16	2,048	200K	6,554M
Transformer	16	2,048	250K	8,192M
Relative Transformer	16	2,048	250K	8,192M

Table 5: Training setups for our neural models on all language pairs.

4.4 Training

We train vanilla phrase-based SMT systems[5] and extract 1000-best lists of unique translations candidates, from which n-gram posteriors are calculated.

All neural models were trained with the Adam optimizer (Kingma and Ba, 2015), dropout (Srivastava et al., 2014), and label smoothing (Szegedy et al., 2016) using the Tensor2Tensor (Vaswani et al., 2018) library. We decode with the average of the last 40 checkpoints (Junczys-Dowmunt et al., 2016a).

We make extensive use of the delayed SGD updates technique we already applied successfully to syntax-based NMT (Saunders et al., 2018). Delaying SGD updates allows to arbitrarily choose the effective batch size even on limited GPU hardware. Large batch training has received some attention in recent research (Smith et al., 2017; Neishi et al., 2017) and has been shown particularly useful for training the Transformer architecture with the Tensor2Tensor framework (Popel and Bojar, 2018). We support these findings in Tab. 4.[6] Our technical infrastructure[7] allows us to train on four P100 GPUs simultaneously, which limits the number of physical GPUs to $g = 4$ and the batch size[8] to $b = 2048$ due to the GPU memory. Thus, the maximum possible effective batch size without delaying SGD updates is $b' = 8192$. Training with delay factor d accumulates gradients over d batches and applies the optimizer update rule on the accumulated gradients. This allows us to scale up the effective number of GPUs to 16 and improve the BLEU score significantly (29.5 vs. 30.3). Note that training regimens are equivalent if their effective batch size is the same, ie. training on 4 physical GPUs with $d = 4$ is mathematically equivalent to training on 16 GPUs without delaying SGD updates. Tab. 5 lists our training setups for the neural architectures used in this work. These training hyper-parameters were chosen empirically. Particularly, we did not find improvements by increasing the number of effective GPUs for SliceNet or longer LSTM training.

We use *news-test2017* as development set on all language pairs to tune the model interpolation weights λ (Eq. 2) and the scaling factor for length normalization.

4.5 Decoding

We use the `beam` search strategy with beam size 8 of the SGNMT decoder (Stahlberg et al., 2017b, 2018) in all our experiments. We apply length normalization (Bahdanau et al., 2015) on German-English and Chinese-English but not on English-German. As outlined in Sec. 2 we either use full posteriors or MBR-style n-gram posteriors from our individual models. SMT n-gram scores are extracted as described by Blackwood et al. (2010) using HiFST's `lmbr` tool. We use SGNMT's `ngram` output format to extract n-gram scores from our neural models.

5 Results

On English-German and German-English *news-test2014* we compute cased BLEU scores with Moses' `multi-bleu.pl` script on tokenized output to be comparable with prior work (Wu et al., 2016; Kaiser et al., 2017; Gehring et al., 2017; Vaswani et al., 2017; Chen et al., 2018). On all other test sets we use `mteval-v13a.pl` to be comparable to the official cased WMT scores.[9]

First, we will discuss our experiments with a single architecture, i.e. single systems and ensembles of two systems with the same architecture. Tab. 6 compares the architectures on all test sets. PBMT as a single system is clearly inferior to all neural systems. Ensembling neural systems helps for all architectures across the board. LSTM

[5] Excluding the UN corpus and the backtranslated data.

[6] We had to reduce the learning rate for $g' = 1$ to avoid training divergence.

[7] http://www.hpc.cam.ac.uk/

[8] We follow Vaswani et al. (2017, 2018) and specify the batch size in terms of number of source and target tokens in a batch, not the number of sentences.

[9] http://matrix.statmt.org/

Architecture	#Systems	English-German				German-English				Chinese-English	
		test14	test15	test16	test17	test14	test15	test16	test17	dev17	test17
PBMT	1	19.6	20.9	25.6	20.0	22.5	27.2	32.6	28.2	14.2	15.8
LSTM	1	27.1	28.8	34.6	28.0	33.8	33.3	40.7	34.8	21.8	22.7
	2	28.2	29.6	35.5	28.5	34.6	34.0	41.4	35.3	22.7	23.6
SliceNet	1	26.8	28.9	33.6	27.6	32.6	32.3	39.8	33.7	21.4	22.5
	2	27.2	29.6	34.6	28.3	33.2	32.9	40.8	34.3	21.8	23.4
R2L Trans.	1	30.3	31.5	36.3	30.2	36.5	35.5	43.5	37.2	24.5	24.9
Transformer	1	30.7	31.9	36.6	30.5	36.7	36.2	43.7	37.9	24.9	25.6
	2	31.1	31.8	37.2	31.0	36.9	36.4	44.0	38.1	26.2	26.2
Rel. Trans.	1	31.2	31.9	37.0	31.1	37.0	36.3	44.1	38.1	24.9	25.8
	2	31.4	32.3	37.7	31.2	37.2	36.5	44.1	38.4	25.1	26.4

Table 6: Single architecture results on all language pairs for single systems and 2-ensembles.

	Full posterior					MBR-based n-gram scores				BLEU (test2017)		
	PBMT	LSTM*	SliceNet*	Trans.	Rel. Trans.	PBMT	LSTM*	SliceNet*	R2L Trans.	en-de	de-en	zh-en
1	✓									20.0	28.2	15.8
2		✓								28.5	35.3	23.6
3			✓							28.3	34.3	23.4
4				✓						30.5	37.9	25.6
5					✓					31.1	38.1	25.8
6				✓	✓					31.3	38.2	26.4
7		✓	✓	✓	✓					31.3	38.2	26.4
8				✓	✓		✓	✓		31.4	38.2	26.6
9				✓	✓		✓	✓	✓	31.4	38.3	26.8
10				✓	✓	✓	✓	✓	✓	31.7	38.7	27.1

Table 7: Model combination with ensembling and MBR.Model scores are weighted with MERT and combined (log-)linearly as described in Sec. 2. *: The LSTM and SliceNet models are 2-ensembles.

is usually slightly better than the convolutional SliceNet, but is much slower to train and decode (cf. Tab. 3). Note that our LSTM 2-ensemble is on par with the best BLEU score in WMT17 (Sennrich et al., 2017), which was also based on recurrent models. Transformer architectures outperform LSTMs and SliceNets on all test sets. The right-to-left Transformer is usually slightly worse, the Transformer with relative positioning slightly better than the standard Transformer setup.

Tab. 7 summarizes our system combination results with multiple architectures. Adding LSTM and SliceNet as full-posterior models to an ensemble of a Transformer and a Relative Transformer does not improve the BLEU score (rows 6 vs. 7). We see very slight improvements when we use these models to extract n-gram scores instead (rows 6 vs. 8). We report further gains by using MBR-based n-gram scores from the right-to-left Transformer and the PBMT system. The improvements from adding PBMT are rather small, but we still found them surprising given that the PBMT baseline is usally more than 10 BLEU points worse than our best single neural model. We list the performance of our submitted systems on all test sets in Tab. 8.

Direction	Test set	BLEU
English-German	news-test14	31.6
	news-test15	32.6
	news-test16	38.5
	news-test17	31.7
	news-test18	46.6
German-English	news-test14	36.8
	news-test15	36.5
	news-test16	45.1
	news-test17	38.7
	news-test18	48.0
Chinese-English	news-dev17	25.7
	news-test17	27.1
	news-test18	27.7

Table 8: BLEU scores of the submitted systems (row 10 in Tab. 7).

6 Related Work

There is a large body of research comparing NMT and SMT (Schnober et al., 2016; Toral and Sánchez-Cartagena, 2017; Koehn and Knowles, 2017; Menacer et al., 2017; Dowling et al., 2018; Bentivogli et al., 2016, 2018). Most studies have found superior overall translation quality of NMT models in most settings, but complementary strengths of both paradigms. Therefore, the literature about hybrid NMT-SMT sys-

tems is also vast, ranging from rescoring and reranking methods (Neubig et al., 2015; Stahlberg et al., 2016; Khayrallah et al., 2017; Grundkiewicz and Junczys-Dowmunt, 2018; Avramidis et al., 2016; Marie and Fujita, 2018), MBR-based formalisms (Stahlberg et al., 2017a, 2018; Iglesias et al., 2018), NMT assisting SMT (Junczys-Dowmunt et al., 2016b; Du and Way, 2017), and SMT assisting NMT (Niehues et al., 2016; He et al., 2016; Long et al., 2016; Wang et al., 2017; Dahlmann et al., 2017; Zhou et al., 2017). We confirm the potential of hybrid systems by reporting gains on top of very strong neural ensembles.

Ensembling is a well-known technique in NMT to improve system performance. However, ensembles usually consist of multiple models of the same architecture. In this paper, we compare and combine three very different architectures (recurrent, convolutional, and self-attention based) in two different ways (full posterior and MBR-based), and find that combination with MBR-based n-gram scores is superior.

7 Conclusion

We have described our WMT18 submission, which achieves very competitive BLEU scores on all three language pairs (English-German, German-English, and Chinese-English) and significantly higher accuracies in a variety of linguistic phenomena compared to other submissions (Avramidis et al., 2018). Our system combines three different neural architecture with a traditional PBMT system. We showed that our MBR-based scheme is effective to combine these diverse models of translation, and that adding the PBMT system to the mix of neural models still yields gains although it is much worse as stand-alone system.

Acknowledgments

This work was supported in part by the U.K. Engineering and Physical Sciences Research Council (EPSRC grant EP/L027623/1).

References

Karim Ahmed, Nitish Shirish Keskar, and Richard Socher. 2017. Weighted transformer network for machine translation. *arXiv preprint arXiv:1711.02132*.

Eleftherios Avramidis, Vivien Macketanz, Aljoscha Burchardt, Jindrich Helcl, and Hans Uszkoreit. 2016. Deeper machine translation and evaluation for German. In *Proceedings of the 2nd Deep Machine Translation Workshop*, pages 29–38. ÚFAL MFF UK.

Eleftherios Avramidis et al. 2018. Fine-grained evaluation of German-English machine translation based on a test suite. In *Proceedings of the Third Conference on Machine Translation, Volume 3*, Brussels, Belgium. Association for Computational Linguistics.

Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E Hinton. 2016. Layer normalization. *arXiv preprint arXiv:1607.06450*.

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)*, Toulon, France.

Luisa Bentivogli, Arianna Bisazza, Mauro Cettolo, and Marcello Federico. 2016. Neural versus phrase-based machine translation quality: a case study. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 257–267. Association for Computational Linguistics.

Luisa Bentivogli, Arianna Bisazza, Mauro Cettolo, and Marcello Federico. 2018. Neural versus phrase-based MT quality: An in-depth analysis on English–German and English–French. *Computer Speech & Language*, 49:52–70.

Graeme Blackwood, Adrià Gispert, and William Byrne. 2010. Efficient path counting transducers for minimum Bayes-risk decoding of statistical machine translation lattices. In *Proceedings of the ACL 2010 Conference Short Papers*, pages 27–32. Association for Computational Linguistics.

Mia Xu Chen, Orhan Firat, Ankur Bapna, Melvin Johnson, Wolfgang Macherey, George Foster, Llion Jones, Niki Parmar, Mike Schuster, Zhifeng Chen, et al. 2018. The best of both worlds: Combining recent advances in neural machine translation. *arXiv preprint arXiv:1804.09849*.

Lonnie Chrisman. 1991. Learning recursive distributed representations for holistic computation. *Connection Science*, 3(4):345–366.

Leonard Dahlmann, Evgeny Matusov, Pavel Petrushkov, and Shahram Khadivi. 2017. Neural machine translation leveraging phrase-based models in a hybrid search. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1411–1420. Association for Computational Linguistics.

Meghan Dowling, Teresa Lynn, Alberto Poncelas, and Andy Way. 2018. SMT versus NMT: Preliminary comparisons for Irish. *Technologies for MT of Low Resource Languages (LoResMT 2018)*, page 12.

Jinhua Du and Andy Way. 2017. Neural pre-translation for hybrid machine translation. *In Proceedings of MT Summit XVI*, 1:27–40.

Mikel L. Forcada and Ramón P. Ñeco. 1997. Recursive hetero-associative memories for translation. In *Biological and Artificial Computation: From Neuroscience to Technology*, pages 453–462, Berlin, Heidelberg. Springer Berlin Heidelberg.

Jonas Gehring, Michael Auli, David Grangier, Denis Yarats, and Yann N Dauphin. 2017. Convolutional sequence to sequence learning. *ArXiv e-prints*.

Roman Grundkiewicz and Marcin Junczys-Dowmunt. 2018. Near human-level performance in grammatical error correction with hybrid machine translation. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. Association for Computational Linguistics.

Hany Hassan, Anthony Aue, Chang Chen, Vishal Chowdhary, Jonathan Clark, Christian Federmann, Xuedong Huang, Marcin Junczys-Dowmunt, William Lewis, Mu Li, et al. 2018. Achieving human parity on automatic Chinese to English news translation. *arXiv preprint arXiv:1803.05567*.

Wei He, Zhongjun He, Hua Wu, and Haifeng Wang. 2016. Improved neural machine translation with SMT features. In *AAAI*, pages 151–157.

Gonzalo Iglesias, William Tambellini, Adrià de Gispert, Eva Hasler, and Bill Byrne. 2018. Accelerating NMT batched beam decoding with LMBR posteriors for deployment. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. Association for Computational Linguistics.

Marcin Junczys-Dowmunt, Tomasz Dwojak, and Hieu Hoang. 2016a. Is neural machine translation ready for deployment? A case study on 30 translation directions. In *Proceedings of the 9th International Workshop on Spoken Language Translation (IWSLT)*, Seattle, WA.

Marcin Junczys-Dowmunt, Tomasz Dwojak, and Rico Sennrich. 2016b. 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: Volume 2, Shared Task Papers*, pages 319–325. Association for Computational Linguistics.

Lukasz Kaiser, Aidan N Gomez, and Francois Chollet. 2017. Depthwise separable convolutions for neural machine translation. *arXiv preprint arXiv:1706.03059*.

Nal Kalchbrenner and Phil Blunsom. 2013. Recurrent continuous translation models. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1700–1709, Seattle, Washington, USA. Association for Computational Linguistics.

Huda Khayrallah, Gaurav Kumar, Kevin Duh, Matt Post, and Philipp Koehn. 2017. Neural lattice search for domain adaptation in machine translation. In *Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 20–25. Asian Federation of Natural Language Processing.

Jack Kiefer. 1953. Sequential minimax search for a maximum. *Proceedings of the American mathematical society*, 4(3):502–506.

Diederik P Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *Proceedings of the International Conference on Learning Representations (ICLR)*.

Philipp Koehn and Rebecca Knowles. 2017. Six challenges for neural machine translation. In *Proceedings of the First Workshop on Neural Machine Translation*, pages 28–39. Association for Computational Linguistics.

Shankar Kumar and William Byrne. 2004. Minimum Bayes-risk decoding for statistical machine translation. In *Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics: HLT-NAACL 2004*.

Aodong Li, Shiyue Zhang, Dong Wang, and Thomas Fang Zheng. 2017. Enhanced neural machine translation by learning from draft. In *Proceedings of APSIPA Annual Summit and Conference*, volume 2017, pages 12–15.

Peng Li, Yang Liu, and Maosong Sun. 2013. Recursive autoencoders for ITG-based translation. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 567–577, Seattle, Washington, USA. Association for Computational Linguistics.

Lemao Liu, Masao Utiyama, Andrew Finch, and Eiichiro Sumita. 2016. Agreement on target-bidirectional neural machine translation. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 411–416. Association for Computational Linguistics.

Zi Long, Takehito Utsuro, Tomoharu Mitsuhashi, and Mikio Yamamoto. 2016. Translation of patent sentences with a large vocabulary of technical terms using neural machine translation. In *Proceedings of the 3rd Workshop on Asian Translation (WAT2016)*, pages 47–57. The COLING 2016 Organizing Committee.

Minh-Thang Luong, Eugene Brevdo, and Rui Zhao. 2017. Neural machine translation (seq2seq) tutorial. *https://github.com/tensorflow/nmt*.

Thang Luong, Hieu Pham, and Christopher D. Manning. 2015. Effective approaches to attention-based neural machine translation. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1412–1421, Lisbon, Portugal. Association for Computational Linguistics.

Wolfgang Macherey, Franz Och, Ignacio Thayer, and Jakob Uszkoreit. 2008. Lattice-based minimum error rate training for statistical machine translation. In *Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing*, pages 725–734. Association for Computational Linguistics.

Benjamin Marie and Atsushi Fujita. 2018. A smorgasbord of features to combine phrase -based and neural machine translation. In *Proceedings of AMTA Workshop on MT Research and the Translation Industry*, Boston, US.

Mohamed-Amine Menacer, David Langlois, Odile Mella, Dominique Fohr, Denis Jouvet, and Kamel Smaïli. 2017. Is statistical machine translation approach dead? In *ICNLSSP 2017-International Conference on Natural Language, Signal and Speech Processing*.

Shuyo Nakatani. 2010. Language detection library for Java.

Masato Neishi, Jin Sakuma, Satoshi Tohda, Shonosuke Ishiwatari, Naoki Yoshinaga, and Masashi Toyoda. 2017. A bag of useful tricks for practical neural machine translation: Embedding layer initialization and large batch size. In *Proceedings of the 4th Workshop on Asian Translation (WAT2017)*, pages 99–109.

Graham Neubig, Makoto Morishita, and Satoshi Nakamura. 2015. Neural reranking improves subjective quality of machine translation: NAIST at WAT2015. In *WAT*, Kyoto, Japan.

Jan Niehues, Eunah Cho, Thanh-Le Ha, and Alex Waibel. 2016. Pre-translation for neural machine translation. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 1828–1836. The COLING 2016 Organizing Committee.

Franz Josef Och. 2003. Minimum error rate training in statistical machine translation. In *Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics*.

Jordan B. Pollack. 1990. Recursive distributed representations. *Artificial Intelligence*, 46(1):77 – 105.

Martin Popel and Ondřej Bojar. 2018. Training tips for the Transformer model. *arXiv preprint arXiv:1804.00247*.

Michael JD Powell. 1964. An efficient method for finding the minimum of a function of several variables without calculating derivatives. *The computer journal*, 7(2):155–162.

Danielle Saunders, Felix Stahlberg, Adrià de Gispert, and Bill Byrne. 2018. Multi-representation ensembles and delayed SGD updates improve syntax-based NMT. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics. To appear.

Carsten Schnober, Steffen Eger, Erik-Lân Do Dinh, and Iryna Gurevych. 2016. Still not there? Comparing traditional sequence-to-sequence models to encoder-decoder neural networks on monotone string translation tasks. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 1703–1714. The COLING 2016 Organizing Committee.

Rico Sennrich, Alexandra Birch, Anna Currey, Ulrich Germann, Barry Haddow, Kenneth Heafield, Antonio Valerio Miceli Barone, and Philip Williams. 2017. The University of Edinburgh's neural MT systems for WMT17. In *Proceedings of the Second Conference on Machine Translation, Volume 2: Shared Task Papers*, pages 389–399, Copenhagen, Denmark. Association for Computational Linguistics.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016a. Improving neural machine translation models with monolingual data. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 86–96. Association for Computational Linguistics.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016b. Neural machine translation of rare words with subword units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1715–1725. Association for Computational Linguistics.

Peter Shaw, Jakob Uszkoreit, and Ashish Vaswani. 2018. Self-attention with relative position representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. Association for Computational Linguistics.

Samuel L Smith, Pieter-Jan Kindermans, and Quoc V Le. 2017. Don't decay the learning rate, increase the batch size. *arXiv preprint arXiv:1711.00489*.

Richard Socher, Jeffrey Pennington, Eric H. Huang, Andrew Y. Ng, and Christopher D. Manning. 2011. Semi-supervised recursive autoencoders for predicting sentiment distributions. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*, pages 151–161, Edinburgh, Scotland, UK. Association for Computational Linguistics.

Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. 2014. Dropout: A simple way to prevent neural networks from overfitting. *The Journal of Machine Learning Research*, 15(1):1929–1958.

Felix Stahlberg, Adrià de Gispert, Eva Hasler, and Bill Byrne. 2017a. Neural machine translation by minimising the Bayes-risk with respect to syntactic translation lattices. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, pages 362–368. Association for Computational Linguistics.

Felix Stahlberg, Eva Hasler, Danielle Saunders, and Bill Byrne. 2017b. SGNMT – A flexible NMT decoding platform for quick prototyping of new models and search strategies. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 25–30. Association for Computational Linguistics.

Felix Stahlberg, Eva Hasler, Aurelien Waite, and Bill Byrne. 2016. Syntactically guided neural machine translation. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 299–305. Association for Computational Linguistics.

Felix Stahlberg, Danielle Saunders, Gonzalo Iglesias, and Bill Byrne. 2018. Why not be versatile? Applications of the SGNMT decoder for machine translation. In *Proceedings of AMTA Workshop on MT Research and the Translation Industry*, Boston, US.

Ilya Sutskever, Oriol Vinyals, and Quoc V Le. 2014. Sequence to sequence learning with neural networks. In *Advances in Neural Information Processing Systems 27*, pages 3104–3112. Curran Associates, Inc.

Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. 2016. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2818–2826.

Antonio Toral and Víctor M. Sánchez-Cartagena. 2017. A multifaceted evaluation of neural versus phrase-based machine translation for 9 language directions. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, pages 1063–1073. Association for Computational Linguistics.

Roy Tromble, Shankar Kumar, Franz Och, and Wolfgang Macherey. 2008. Lattice minimum Bayes-risk decoding for statistical machine translation. In *Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing*, pages 620–629. Association for Computational Linguistics.

Ashish Vaswani, Samy Bengio, Eugene Brevdo, Francois Chollet, Aidan N Gomez, Stephan Gouws, Llion Jones, Łukasz Kaiser, Nal Kalchbrenner, Niki Parmar, et al. 2018. Tensor2tensor for neural machine translation. In *Proceedings of AMTA Workshop on MT Research and the Translation Industry*, Boston, US.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems 30*, pages 6000–6010. Curran Associates, Inc.

Xing Wang, Zhengdong Lu, Zhaopeng Tu, Hang Li, Deyi Xiong, and Min Zhang. 2017. Neural machine translation advised by statistical machine translation. In *AAAI*, pages 3330–3336.

Yonghui Wu, Mike Schuster, Zhifeng Chen, Quoc V Le, Mohammad Norouzi, Wolfgang Macherey, Maxim Krikun, Yuan Cao, Qin Gao, Klaus Macherey, et al. 2016. Google's neural machine translation system: Bridging the gap between human and machine translation. *arXiv preprint arXiv:1609.08144*.

Long Zhou, Wenpeng Hu, Jiajun Zhang, and Chengqing Zong. 2017. Neural system combination for machine translation. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 378–384. Association for Computational Linguistics.