NMT: The Story till now...

Liling Tan

December 4, 2018

1 Machine Translation Systems

Neural Machine Translation (NMT) has surpassed the performance of the Statistical Machine Translation (SMT) paradigm[1]. SMT is grounded upon information theory where multiple component models[Brown et al., 1990, Chiang, 2005, Och & Ney, 2002] are trained independently and combined using in a log-linear fashion, the weighs of each component is trained using various tuning algorithm[Chiang, 2012, Hopkins & May, 2011, Och, 2003]. Conversely, NMT builds on the work of learning representation of language using neural networks[Bengio et al., 2003, Collobert et al., 2011, Mikolov et al., 2013, Schwenk, 2012] to jointly train a model that maps the source language neural representation to a target language representation.

2 The Tsunami Bellows

While deep neural net is good at learning representation for visual image recognition given the fixed sized pixel inputs, the initial challenge in applying deep neural networks to language is finding the solution to 'sensibly encode' variable-length input to a fixed dimensionality vector. Cho et al. [2014b] proposed the Recurrent Neural Net (RNN) Encoder-Decoder framework that consists of a Long Short Term Memory (LSTM) [Hochreiter & Schmidhuber, 1997] encoder that maps the variable length source language sequence to a fixed length vector and a LSTM decoder that takes that fixed length vector as the input and maps it to a variable-length target sequence. The encoder and decoder networks are trained jointly to maximize the conditional log-likelihood of the target sequence given the source sequence. Kalchbrenner & Blunsom [2013] proposed a similar Recurrent Continuous Translation Models (RCTM) framework to estimate the conditional probability of the target sequence given the source sequence; however the encoder is using a Convolutional Neural Net (CNN) n-gram sentence model and the decoder is a vanilla RNN model that generates the output sequence in a state-wise manner. While the neural methods proposed for machine translation were novel, they did not outperform the state-of-art PBMT system. The true neural tsunami came when Sutskever et al. [2014] applied the encoder-decoder framework to the Workshop for Machine Translation 2014 (WMT14) dataset and successfully showed that it outperformed Durrani et al. [2014] PBMT system with Operation Sequence Model (OSM). Thereafter, the encoder-decoder framework takes on an alter-name, viz. Sequence-to-Sequence (Seq2Seq) framework.

3 The Two-Trick Pony

The evolution of Seq2Seq frameworks has been rapid since 2014. A natural improvement to the single-layer encoder-decoder network is to extend the network layerwise; deeper stacked networks had showed to work well in various image processing tasks [Ioffe & Szegedy, 2015, Simonyan & Zisserman, 2014]. However, the vanishing/exploding gradients issue hinders the training of deep networks; when deeper network converges, performance saturates and degrades quickly. He et al. [2016] introduced the residual learning that resolves the degradation issue by shortcutting the parameters from previous layers as an identity to deeper layers, making the approximators in the deeper layers to drive the parameter weights towards zero to approach the identity mappings.

[1] Phrase-Based Machine Translation (PBMT) being the most prominent method in SMT

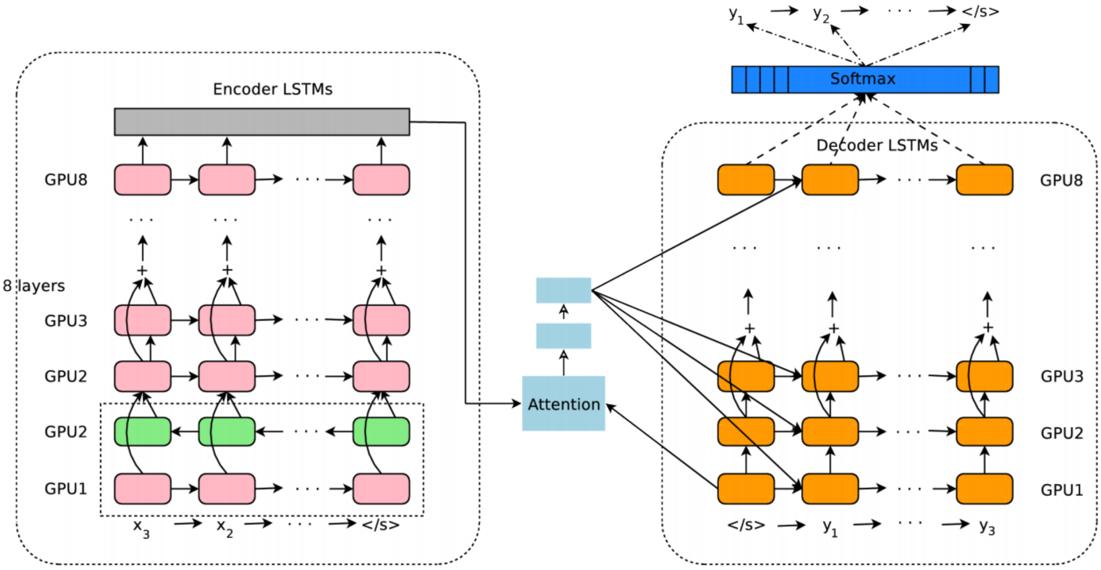

Figure 1: Multi-layered RNN Sequence-to-Sequence Architecture with Residual Connections and Attention Mechanism (Image from [Wu et al., 2016])

Another major advancement is the use of memory mechanism to align the source sequence positions to the target sequence state. Cho et al. [2014a] noted the Seq2Seq model deteriorates quickly as the input sequence length increases, this is because the decoder is forced to make a hard decision to predict a target word at every state. Bahdanau et al. [2014] proposed the attention mechanism [Bazzani et al., 2011, Denil et al., 2012] that allows the Seq2Seq model to focus on a set of positions from the source sentence to form a context vector that are most relevant to the current state in the target sequence. In addition to the previous state outputs, it takes the context vector associated from the attention mechanism to predict the current word. Luong et al. [2015] extends the attention mechanism to introduce global and local attention mechanism. The global attention model infers a variable-length context vector based on the current state and all source states, the context vector is then averaged over all source states. The local attention model predicts a single position from the source sequence at the current target state and the context vector is created for a windowed centered from the predicted position, similarly a weighted average is taken but only over the windowed source states. Most system employed the deeply stacked Seq2Seq networks with attention mechanism to achieve state-of-the-art performance [Barone et al., 2017, Crego et al., 2016, Johnson et al., 2016, Zhou et al., 2016].

4 Stack'em All

Most systems employed the deeply stacked recurrent Seq2Seq networks with attention mechanism to achieve state-of-the-art performance Crego et al. [2016], Johnson et al. [2016], Wu et al. [2016], Zhou et al. [2016]. Britz et al. [2017] explored biunidirectional stacked encoders Seq2Seq models[2] with varying experimental settings on encoder and decoder embedding sizes, RNN cell types, network depths, attention mechanisms and decoder beam size. Although the paper offers new architectural inventions, the work showed that state-of-art machine translation can be achieved with good initialization and tedious hyperparameter tuning. Othogonally, Barone et al. [2017] evaluated several variants of stacking the encoder-decoder networks[3] showing that alternating stacked encoders [Zhou et al., 2016] outperform biunidirectional stacked encoders [Wu et al., 2016]. Extending the Recurrent Highway Network [Zilly et al., 2017] and Recurrent Transition Depth[4] [Firat et al., 2017], Barone et al. [2017] introduced the transition depth encoder-decoder network that replaces the individual GRU cells in the stacked layers with the multi-layered recurrent transition depth

[2] Using Tensorflow [Abadi et al., 2016], https://github.com/google/seq2seq

[3] With Nematus [Sennrich et al., 2017], https://github.com/EdinburghNLP/nematus

[4] https://github.com/nyu-dl/dl4mt-tutorial/blob/master/docs/cgru.pdf

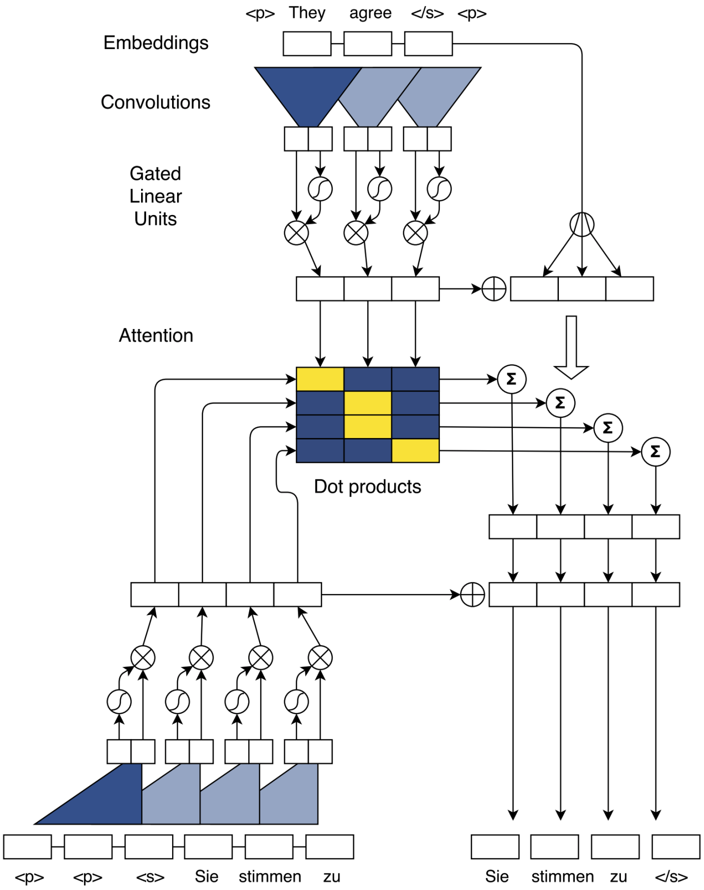

Figure 2: Fully Convolution Sequence-to-Sequence Architecture with Gated Linear Units and Attention Mechanism (Image from [Gehring et al., 2017])

cells. Empirically, based on the Workshop for Machine Translation 2017 (WMT17) dataset, they found that the best performing system (translation accuracy) was trained by alternating between the biunidirectional stacked encoders and the transition depth network, the most efficient system (speed or model size) was achieved with the deep transition architecture.

5 The Convolution Revolution

In addition to stacking recurrent networks, Gehring et al. [2017] extends the Kalchbrenner & Blunsom [2013] CNN-RNN encoder-decoder architecture to a fully CNN architecture[5] to compute the intermediate states in both the encoder and decoder. Compared to RNN, convolutions create fixed-sized contextual representations and the context size can grow larger by stacking multiple layer of convolutions to adapt to the maximum length in the variable-length sequences. Without the constraint of being dependent on the previous hidden states, CNN is easily parallelizable. Multi-layered CNN also creates a hierarchical representation that encodes the n size input in O(n/k) convolution operations for kernels with with, k, compared to linear O(n) for RNN. The CNN Seq2Seq architecture first embeds the input elements and their respective positions in a distributed space. The embeddings are fed into the convolution layer that computes the intermediate states based on a fixed size input sequence. Each layer contains a 1-dimensional convolution followed by an activation/non-linearity. The CNN Seq2Seq architecture in Gehring et al. [2017] uses the Gated Linear Units [Dauphin et al., 2017] which implements a gated mechanism over the convolution outputs. Residual connections between alternate convolution layers are used to enable deep CNN Seq2Seq networks.

[5] Using Pytorch, https://github.com/pytorch/fairseq/tree/master/fairseq

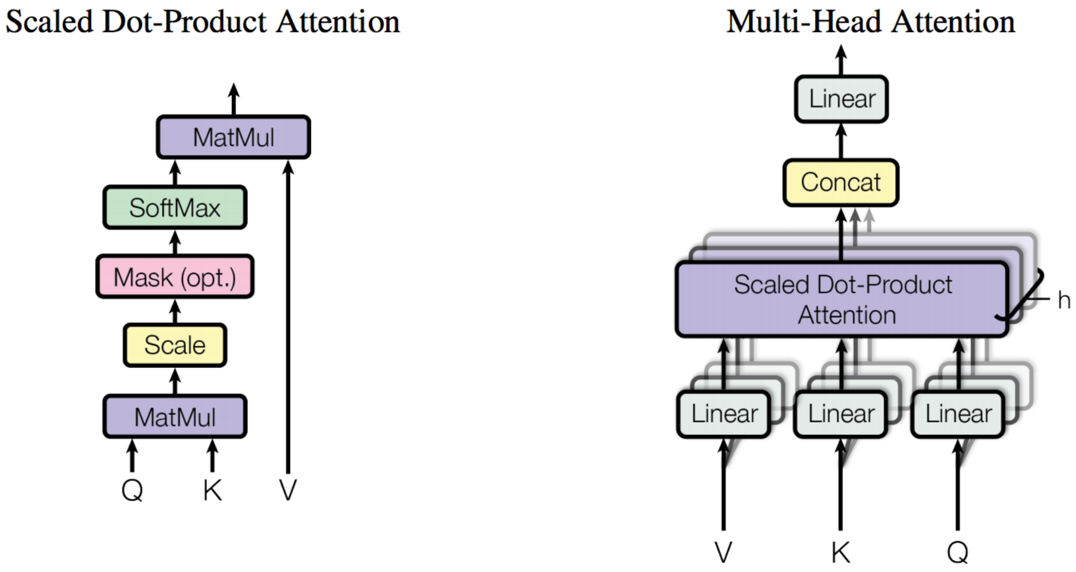

Figure 3: Transformer Sequence-to-Sequence architecture using the multi-headed self-attention (Image from [Vaswani et al., 2017])

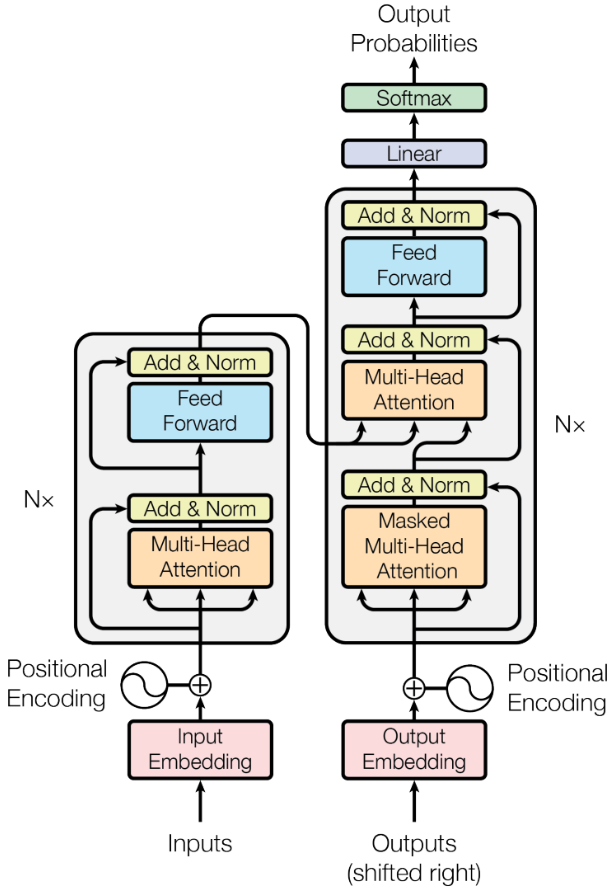

Figure 4: Self-attention mechanism (left) and the Multi-headed Attention (right) (Image from [Vaswani et al., 2017])

6 Throw away your RNNs

While the CNN Seq2Seq model has reduced sequential computation and allow parallelization, the number of operations required to align two arbitrary in-/output grows respective to the distance between the two positions. The Transformer [Vaswani et al., 2017] architecture reduces the number of operations to a constant by introducing the multi-headed self-attention mechanism. The self-attention (i) first creates three vectors of the same dimensions for each state from the input sequence, i.e. the query, key and value vectors[6] and (ii) then takes the dot product of the query and key vector for each word against every other word in the sequence, (iii) the output of all word to word scores from (ii) is then divided by the root of the dimensions of the query, key, value vectors[7] and (iv) for each word take the softmax of scores with respect to the other words, then (v) multiply the value vector to the softmax score from (iv) and finally (vi) sum the weighted valued vector from (v) to produce the output of the self-attention layer.

To extend the ability for a word to 'attend' to different positions in the sequence, Vaswani et al. [2017] introduced the multi-headed attention mechanism which is essential duplicating the create of the query, key and value vectors in the self-attention layer with different weight matrices and concatenating the output of the self-attention layer to form the output of a multi-headed attention layer. The self attention layer is then followed by a full-connected Feed-Forward Network to emulate a single layer of encoder/decoder in a typical unidirectional stacked sequence-to-sequence network. The full transformer architecture follows a standard unidirectional stacked encoder-decoder framework to stack multiple multi-headed and feed-forward layers. Additionally, the same residual shortcutting is applied to the stacked layers.

7 State of Now

To summarize, the state-of-art Sequence-to-Sequence architectures for Neural Machine Translation au current are (a) Recurrent Neural Network with attention mechanism, (b) Convolution Neural Network with attention mechanism and (c) Self-attention Feed-Forward Network.

	RNN with Attention	**CNN with Attention**	**Self-Attention FFN**
Encoder / Decoder	Gated RNN units (e.g. GRU/LSTM)	Convolution kernels	Multi-headed Self-attention + FFN
Residual + Stacked Network	Allowed	Allowed	Allowed
Activation Func	Unspecified (commonly Sigmoid, Tanh, ReLU)	Gated Linear Unit	ReLU

Table 1: State of now

[6] In the original Vaswani et al. [2017] paper, they use 64 as the size for the query, key and value vectors

[7] IN the original Vaswani et al. [2017] paper, it would be root(64) = 8

References

Abadi, Martín, Paul Barham, Jianmin Chen, Zhifeng Chen, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Geoffrey Irving, Michael Isard et al. 2016. Tensorflow: A system for large-scale machine learning .

Bahdanau, Dzmitry, Kyunghyun Cho & Yoshua Bengio. 2014. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473* .

Barone, Antonio Valerio Miceli, Jindřich Helcl, Rico Sennrich, Barry Haddow & Alexandra Birch. 2017. Deep architectures for neural machine translation. In *Proceedings of the second conference on machine translation*, 99–107.

Bazzani, Loris, Hugo Larochelle, Vittorio Murino, Jo-anne Ting & Nando D Freitas. 2011. Learning attentional policies for tracking and recognition in video with deep networks. In *Proceedings of the 28th international conference on machine learning (icml-11)*, 937–944.

Bengio, Yoshua, Réjean Ducharme, Pascal Vincent & Christian Jauvin. 2003. A neural probabilistic language model. *Journal of machine learning research* 3(Feb). 1137–1155.

Britz, Denny, Anna Goldie, Minh-Thang Luong & Quoc Le. 2017. Massive exploration of neural machine translation architectures. In *Proceedings of the 2017 conference on empirical methods in natural language processing*, 1442–1451.

Brown, Peter F, John Cocke, Stephen A Della Pietra, Vincent J Della Pietra, Fredrick Jelinek, John D Lafferty, Robert L Mercer & Paul S Roossin. 1990. A statistical approach to machine translation. *Computational linguistics* 16(2). 79–85.

Chiang, David. 2005. A hierarchical phrase-based model for statistical machine translation. In *Proceedings of the 43rd annual meeting on association for computational linguistics*, 263–270. Association for Computational Linguistics.

Chiang, David. 2012. Hope and fear for discriminative training of statistical translation models. *Journal of Machine Learning Research* 13(Apr). 1159–1187.

Cho, Kyunghyun, Bart Van Merriënboer, Dzmitry Bahdanau & Yoshua Bengio. 2014a. On the properties of neural machine translation: Encoder-decoder approaches. *arXiv preprint arXiv:1409.1259* .

Cho, Kyunghyun, Bart Van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk & Yoshua Bengio. 2014b. Learning phrase representations using rnn encoder-decoder for statistical machine translation. *arXiv preprint arXiv:1406.1078* .

Collobert, Ronan, Jason Weston, Léon Bottou, Michael Karlen, Koray Kavukcuoglu & Pavel Kuksa. 2011. Natural language processing (almost) from scratch. *Journal of Machine Learning Research* 12(Aug). 2493–2537.

Crego, Josep, Jungi Kim, Guillaume Klein, Anabel Rebollo, Kathy Yang, Jean Senellart, Egor Akhanov, Patrice Brunelle, Aurelien Coquard, Yongchao Deng et al. 2016. Systran's pure neural machine translation systems. *arXiv preprint arXiv:1610.05540* .

Dauphin, Yann N, Angela Fan, Michael Auli & David Grangier. 2017. Language modeling with gated convolutional networks. In *International conference on machine learning*, 933–941.

Denil, Misha, Loris Bazzani, Hugo Larochelle & Nando de Freitas. 2012. Learning where to attend with deep architectures for image tracking. *Neural computation* 24(8). 2151–2184.

Durrani, Nadir, Barry Haddow, Philipp Koehn & Kenneth Heafield. 2014. Edinburgh's phrase-based machine translation systems for wmt-14. In *Proceedings of the ninth workshop on statistical machine translation*, 97–104.

Firat, Orhan, Kyunghyun Cho, Baskaran Sankaran, Fatos T Yarman Vural & Yoshua Bengio. 2017. Multi-way, multilingual neural machine translation. *Computer Speech & Language* 45. 236–252.

Gehring, Jonas, Michael Auli, David Grangier, Denis Yarats & Yann N Dauphin. 2017. Convolutional sequence to sequence learning. *arXiv preprint arXiv:1705.03122* .

He, Kaiming, Xiangyu Zhang, Shaoqing Ren & Jian Sun. 2016. Deep residual learning for image recognition. In *Proceedings of the ieee conference on computer vision and pattern recognition*, 770–778.

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

Hopkins, Mark & Jonathan May. 2011. Tuning as ranking. In *Proceedings of the conference on empirical methods in natural language processing*, 1352–1362. Association for Computational Linguistics.

Ioffe, Sergey & Christian Szegedy. 2015. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *arXiv preprint arXiv:1502.03167* .

Johnson, Melvin, Mike Schuster, Quoc V Le, Maxim Krikun, Yonghui Wu, Zhifeng Chen, Nikhil Thorat, Fernanda Viégas, Martin Wattenberg, Greg Corrado et al. 2016. Google's multilingual neural machine translation system: enabling zero-shot translation. *arXiv preprint arXiv:1611.04558* .

Kalchbrenner, Nal & Phil Blunsom. 2013. Recurrent continuous translation models. In *Proceedings of the 2013 conference on empirical methods in natural language processing*, 1700–1709.

Luong, Thang, Hieu Pham & 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*, 1412–1421.

Mikolov, Tomas, Ilya Sutskever, Kai Chen, Greg S Corrado & Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, 3111–3119.

Och, Franz Josef. 2003. Minimum error rate training in statistical machine translation. In *Proceedings of the 41st annual meeting on association for computational linguistics-volume 1*, 160–167. http://www.aclweb.org/anthology/P03-1021.pdf.

Och, Franz Josef & Hermann Ney. 2002. Discriminative training and maximum entropy models for statistical machine translation. In *Proceedings of the 40th annual meeting of the association for computational linguistics, july 6-12, 2002, philadelphia, pa, USA.*, 295–302.

Schwenk, Holger. 2012. Continuous space translation models for phrase-based statistical machine translation. *Proceedings of COLING 2012: Posters* 1071–1080.

Sennrich, Rico, Orhan Firat, Kyunghyun Cho, Alexandra Birch, Barry Haddow, Julian Hitschler, Marcin Junczys-Dowmunt, Samuel Läubli, Antonio Valerio Miceli Barone, Jozef Mokry et al. 2017. Nematus: a toolkit for neural machine translation. *arXiv preprint arXiv:1703.04357* .

Simonyan, Karen & Andrew Zisserman. 2014. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556* .

Sutskever, Ilya, Oriol Vinyals & Quoc V Le. 2014. Sequence to sequence learning with neural networks. In Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence & K. Q. Weinberger (eds.), *Advances in neural information processing systems 27*, 3104–3112. Curran Associates, Inc. http://papers.nips.cc/paper/5346-sequence-to-sequence-learning-with-neural-networks.pdf.

Vaswani, Ashish, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser & Illia Polosukhin. 2017. Attention is all you need. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan & R. Garnett (eds.), *Advances in neural information processing systems 30*, 5998–6008. Curran Associates, Inc. http://papers.nips.cc/paper/7181-attention-is-all-you-need.pdf.

Wu, Yonghui, 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* .

Zhou, Jie, Ying Cao, Xuguang Wang, Peng Li & Wei Xu. 2016. Deep recurrent models with fast-forward connections for neural machine translation. *Transactions of the Association of Computational Linguistics* 4(1). 371–383.

Zilly, Julian, Rupesh Srivastava, Jan Koutnik & Jürgen Schmidhuber. 2017. Recurrent highway networks. In *Proceedings of the 34 th international conference on machine learning*, vol. 70, 4189–4198.