Machine Translation (2021) 35:289–299
https://doi.org/10.1007/s10590-021-09277-x

BOOK REVIEW

Philipp Koehn: Neural Machine Translation

Cambridge University Press, 30 Jun 2020, www.cambridge.org/9781108497329, DOI: 10.1017/9781108608480

Wandri Jooste[1] · **Rejwanul Haque**[2] · **Andy Way**[1]

Published online: 9 July 2021

Abstract
Neural machine translation (NMT) is an approach to machine translation (MT) that uses deep learning techniques, a broad area of machine learning based on deep artificial neural networks (NNs). The book *Neural Machine Translation* by Philipp Koehn targets a broad range of readers including researchers, scientists, academics, advanced undergraduate or postgraduate students, and users of MT, covering wider topics including fundamental and advanced neural network-based learning techniques and methodologies used to develop NMT systems. The book demonstrates different linguistic and computational aspects in terms of NMT with the latest practices and standards and investigates problems relating to NMT. Having read this book, the reader should be able to formulate, design, implement, critically assess and evaluate some of the fundamental and advanced deep learning techniques and methods used for MT. Koehn himself notes that he was somewhat overtaken by events, as originally this book was envisaged only as a chapter in a revised, extended version of his 2009 book *Statistical Machine Translation*. However, in the interim, NMT completely overtook this previously dominant paradigm, and this new book is likely to serve as the reference of note for the field for some time to come, despite the fact that new techniques are coming onstream all the time.

✉ Andy Way
andy.way@adaptcentre.ie

Wandri Jooste
wandri.jooste@adaptcentre.ie

Rejwanul Haque
rejwanul.haque@adaptcentre.ie

[1] ADAPT Centre, School of Computing, Dublin City University, Dublin, Ireland

[2] School of Computing, National College of Ireland, Dublin, Ireland

1 Part I Introduction

Part I of the book serves as an introduction to MT in general and NMT in particular. In four chapters, it covers some of the important issues and concepts, how MT is used today, the history of MT and how MT can be evaluated.

The short first chapter entitled "The Translation Problem" begins with examples as to why MT can be so difficult. It lists some of the Natural Language Processing (NLP) abstractions to have emerged over the years which have proven useful to explain these difficulties. It then motivates the switch away from rule-based methods to data-driven techniques, but notes that these methods cannot match human performance. Yet! The chapter ends with a brief introduction to datasets, toolkits and evaluation campaigns.

The second chapter discusses uses of MT and explains why the current aim of MT is not to achieve perfect translation, but rather to improve upon current levels of quality. MT allows quick access to information, where the goal is to get an overview of what a given piece of content is about, rather than having it perfectly translated. Professional translators are needed when high-quality translations of documents are required, but using MT can greatly increase their productivity by post-editing the automatic output. These post-editing techniques work much better in a collaborative platform where the MT models are adaptive and interactive. MT has been and is currently being used in various communication areas, such as speech (e.g. Skype Translator)[1] and sign language translation. MT is also shown to be an important part of NLP pipelines such as crosslingual information retrieval and extraction, and multimodal MT.

Chapter 3 sketches the history of MT and how it developed alongside neural network-based machine learning techniques. It starts with how artificial NNs are inspired by our own biological processing. It then explains how in the 1980s the field started to move away from rule-based methods towards data-driven MT techniques such as example-based MT (Carl and Way 2003) and statistical MT (SMT) (Koehn 2009), especially with the rise in open-source software and the availability of parallel corpora across many language-pairs. As for the introduction of neural approaches to MT, at first neural language models were incorporated into SMT systems and later on NNs were used to extend phrase translation tables and reordering models. Finally, MT systems were trained with the use of NN-based end-to-end learning protocols and with the addition of the attention mechanism the same (if not better) translation quality could be achieved with NMT systems as with SMT. Eventually, with the introduction of new and improved techniques, neural models have superseded the previously dominant SMT approach, to the extent that they can be considered the state-of-the-art, especially in terms of research in MT, but increasingly in industrial pipelines too.

The last chapter of Part I, Chapter 4, centres around how to measure the quality of MT outputs (Way 2018). There are evaluation metrics in place to track the

[1] https://www.skype.com/en/features/skype-translator/.

quality of translation, but it is still an open research question, especially in light of the shift to NMT. The first type of evaluation Koehn describes is task-based evaluation, where the quality emanating from MT is 'good enough' for a number of real-world problems to not have to warrant commissioning a full manual translation. A key use-case in task-based translation is evaluating the productivity of human translators when post-editing MT output rather than translating from scratch. Next, human assessments are discussed and they are not as informative as task-based evaluation, but are cheaper and can scale better. The evaluation is (or should be, cf. Läubli et al. (2018), Toral et al. (2018)) performed by professional translators who either grade MT system translations according to fluency and adequacy, or rank two systems against one another. Another way of doing this is using crowd-sourcing, where participants score sentences that are a random mix of system translations, human-generated translations and perhaps degraded (i.e. replacing a phrase with a random translation from elsewhere) versions of the system's translations. Finally, automatic evaluation metrics are discussed. Human assessments are a lot more accurate than automatic evaluation methods, but are slow and costly so cannot be used as frequently as one might like. In this part of the chapter BLEU (Papineni et al. 2002), METEOR (Banerjee and Lavie 2005), TER (and HTER) (Snover et al. 2006) and characTER (Wang et al. 2016) are described and compared to one another in terms of advantages and disadvantages. The last section of this chapter outlines all the shortcomings of automatic evaluation metrics, especially BLEU since it is most commonly used, and describes methods for evaluating evaluation metrics, some of which require additional resources. Some of the more recently proposed automatic evaluation metrics are also briefly covered.

2 Part II Basics

Part II covers all of the main concepts underpinning NMT, such as model architecture and core aspects of training and decoding. Each chapter is concluded with a hands-on example related to the contents of the specific chapter and a list of further reading relating to each topic.

Chapter 5, the first chapter of Part II, takes a deeper look at neural networks. Koehn starts off by describing linear models in terms of input nodes, weights, output nodes and tuning. A translation example is then used to show how these concepts work together. Koehn points out that these models cannot handle nonlinear relationships between features as well as the fact that the features used in MT are not always linearly separable. Next, Koehn goes on to explain how NNs modify linear models, the most significant changes being the use of multiple layers, hidden layers between input and output layers, and a nonlinear activation function. These NNs are trained by repeatedly feeding training examples into the network and updating the weights according to the output. The formulas for updating weights when using back-propagation and gradient descent are derived and explained in detail, and a short example is provided to show how weights are updated. Furthermore, Koehn explains why a validation set is necessary to avoid overfitting of the data. The last part of the chapter

focuses on exploiting parallel processing and using graphic processing units (GPUs) for calculations consisting of vector and matrix operations in NNs.

Chapter 6 "Computation Graphs" describes toolkits that can be used to define NNs and how they work. The mathematical equations that describe NNs can be represented as computation graphs. These graphs can be any type of acyclical directed graph. The same feed-forward network that was used in previous chapters as an example is used here to illustrate what a computation graph looks like and how it is used. The next part of the chapter explains how computation graphs simplify model training in two ways in terms of gradient computations. The first simplification occurs when calculating the error. This is done by adding an error function, like the commonly used L2 norm, to the end of the computation graph. The second occurs when updating parameters through the use of back-propagation. This is done by updating the gradients of nodes with respect to each input to that node and the method for updating these gradients is explained with a detailed step-by-step example.

Chapter 7 explains the statistics and workings behind neural language models, how to represent words and how NNs can be used to model probability distributions of language models. Different methods of representing words as vectors are mentioned, but one-hot encoding vectors are introduced in more detail as is the use of embedding matrices to create word embeddings. Next, the architecture of feed-forward neural network language models is described in the context of one-hot vectors, word embeddings, and the training process. Koehn then describes the concept of distributional lexical semantics and the importance of using such representations to define words by their distributional properties, as well as the theory behind noise contrastive estimation as a method to train a self-normalising model. The last type of language model that is introduced is the recurrent neural network (RNN) language model with long short-term memory (LSTM) units. The differences and similarities between the RNN and feed-forward language models are illustrated and some of the weaknesses of using an RNN model are addressed. This is done through the use of an LSTM unit and a gated recurrent unit (GRU) as a simplification of the LSTM cell. The final sections of this chapter explain the architectures of deep models.

Chapter 8 covers neural translation models as an extension of the recurrent neural language model, which includes the addition of an alignment model. The architecture of the encoder and decoder of a neural translation model are described through the theory behind these architectures, and then the attention mechanism (to model alignment) is introduced in addition to describing how this mechanism connects the encoder and decoder. Training a neural language model requires a more detailed explanation than simply describing the model architectures. Koehn points out that the number of training steps in the encoder and decoder differ for each training example, and sentences of different lengths require different computation graphs. This part of the chapter concentrates on introducing training-specific concepts such as parallelisation over multiple GPUs, maxi- and mini-batches and a summary of the steps taken when training a neural translation model.

Chapter 9 takes a deeper look at how the decoder of a deep model searches for the best output sequences via the use of beam search, and describes various extensions of the decoding algorithm mentioned in the previous chapter including a number

of optimisation methods. Ensemble decoding is briefly introduced and then the the different methods for implementing reranking decoding—an extension of ensemble decoding—is described in more detail. One of the main desired outcomes of these methods is to create a more diverse n-best list of translation candidates. Although reranking is described in detail, Koehn notes that these methods are (so far) rarely implemented in NMT systems despite being a core element in SMT. The last concept presented in this chapter is directed decoding, which sets constraints for the decoding algorithm. Directed decoding was implemented relatively easily in SMT, but implementing it in NMT is much more difficult and is still an open research question. Two methods that have resulted in relatively successful outcomes, namely grid search and enforcing attention, are discussed.

3 Part III Refinements

Part III of the book is focused on training methods for state-of-the-art NMT models, along with open research questions and unsolved problems. Once again each chapter is concluded with an extensive list of further readings on related topics.

In Chapter 10, Koehn describes various machine learning pitfalls and presents methods used in NMT to address these problems. Most of these issues occur due to gradient descent training in terms of the learning rate, weight initialisation and handling local optima. The first method Koehn discusses to remedy some of these isues is the importance of randomness when initialising weights and also shuffling the training data before model training starts. Next, setting the optimal learning rate is addressed, and Koehn observes that methods which alter the learning rate as training proceeds typically work best. One of the most difficult problems concerning optimisation is ensuring that the model converges to the global optimum, or close to it, rather than a local optimum. Although there is no exact solution to this problem, Koehn mentions some of the methods that generally work well, including regularisation (Zaremba et al. 2014), curriculum learning (Zhang et al. 2018) and dropout (Gal and Ghahramani 2016). Finally the problem of vanishing and exploding gradients is addressed, followed by sentence-level optimisation techniques and the benefits of using these methods compared to word-level optimisation.

Throughout the previous chapters in this book Koehn looked at NMT models based on recurrent NNs. In Chapter 11 he introduces different NN architectures that can be implemented to create MT models, the most important of which are different attention mechanisms. The first difference in architecture that is discussed is the use of factored decompositions, in which a smaller vocabulary is used for word embeddings to reduce the dimensions of the representations. Furthermore, various mathematical operations are provided that can be implemented to reduce the size of representations or combine multiple inputs. A quick overview of convolutional NNs (CNNs) (Kalchbrenner et al. 2014) is given, followed by different architectures that make use of attention models. Koehn begins this part of the chapter by introducing the concepts of multihead attention (Vaswani et al. 2017), fine-grained attention (Choi et al. 2018) and self-attention (Vaswani et al. 2017) when using attention mechanisms. The first end-to-end NMT model is described which was based on the

use of CNNs. This type of architecture is explained in more detail, but it is noted that these models did not achieve competitive results compared to SMT, which led to the combination of CNNs with attention. Koehn explains how the encoder, decoder and attention mechanism are implemented, as well as the main differences compared to the NMT model used as an example in previous chapters. Finally, the self-attention components of the Transformer model are described. This includes the self-attention layer to encode an input sentence and the self-attention, attention and feed-forward sublayers in the decoder.

Chapter 12 takes a deeper look at representing sentences and words, like the use of vocabularies and methods for breaking sentences up into words, as well as breaking words up into characters. The first part of this chapter describes various methods to train general-purpose word embeddings and how they differ from one another. These training methods include latent semantic analysis, continuous bag-of-words (Mikolov et al. 2013), skip-gram (Mikolov et al. 2013), GloVe (Pennington et al. 2014), ELMo (Peters et al. 2018) and BERT (Devlin et al. 2019). The basic ideas of multilingual word embeddings are briefly discussed based on the comprehensive survey published by Ruder et al. (2019).[2] The handling of vocabularies of languages is a vital component of word embeddings for neural methods, since they are extremely large and very unevenly distributed. The concept of rare words, in terms of how they were treated initially, and restricting vocabulary sizes for neural translation models is introduced. Currently, a common approach to handle rare words is to break them up into subword units, such as commonly used technique of byte pair encoding (Sennrich et al. 2016a) and its sentence piece (Kudo and Richardson 2018) variant which can be used to avoid tokenisation and other preprocessing steps. The expectation maximisation algorithm and subword regularisation training refinements for these approaches are also discussed. The final methods that are discussed in this chapter are character-based models (Ling et al. 2015; Chung et al. 2016) which are used to break words up into characters and the contrasts between subword models and character-based models are pointed out. Although Koehn only discusses the most commonly used embedding methods, the further reading list is particularly extensive in this chapter and offers more methods for the reader to consider.

In general, MT systems need to be adapted to a specific domain to achieve the best performance on a given task. Chapter 13 briefly defines a domain in terms of its application in MT and points out why adaption techniques require extensive experimentation in order to obtain optimal results, before introducing various specific domain adaptation techniques. The first type of adaptation method that Koehn introduces are mixture models which can either be implemented by combining in-domain and out-of-domain training data to train a combined domain model, or by training separate models on in-domain and out-of-domain data, respectively, and then combining the models. He then shows how these methods can be implemented in a multi-domain scenario to create a single neural architecture, and ends with a description of topic models to automatically cluster sentence pairs in a corpus based on input sentences. Next, methods for choosing which out-of-domain data matches

[2] Koehn refers to the 2017 survey which has since been revised and published in 2019.

in-domain data when using subsampling for domain adaptation are described. Lastly, the fine-tuning adaptation methods are introduced and various considerations to take into account when experimenting with fine-tuning are discussed. Applications of fine-tuning for document-level adaptation (Kothur et al. 2018), sentence-level adaptation (Farajian et al. 2017) and curriculum adaption (van der Wees et al. 2017) are also described. As for further reading in this chapter, Koehn once again provides an extensive list. While most of this literature is primarily related to SMT, it is still relevant to NMT, so new PhD students in the field searching for a topic might profit from reading these papers.

Chapter 14 concentrates on methods that are useful when parallel corpora are limited, by using various machine learning methods in the context of MT, such as unsupervised learning, transfer learning and multitask learning. This chapter consists of three parts, in which the first discusses the use of monolingual data, the second considers the use of multiple language pairs and the third describes training models on related tasks. Monolingual data can be used in two ways: (i) by training a language model which can be integrated into an NMT architecture; and (ii) by creating synthetic data through the use of back-translation (Sennrich et al. 2016b; Poncelas et al. 2018) and round trip training (He et al. 2016). When using multiple language pairs, different scenarios are described including the use of multiple input languages, multiple output languages or sharing components between models. Many of the core components of MT are shared with various other NLP tasks. In this section Koehn explains how these components can be used to train general systems which can perform different NLP tasks. Pretrained word embeddings or pretrained encoders and decoders can be used to initialise training of translation models. In contrast, the use of multitask training to train a combined model on various NLP tasks, which concludes training by focusing on a specific task, is discussed.

Chapter 15 covers the use of linguistic research and structures in MT. The first method that is described briefly is guided alignment training, where alignments that have been precomputed using the same means as in SMT are added to the training data. Next coverage models (Tu et al. 2016) are introduced and a method for integrating them into an NMT model is described, followed by the augmentation of a coverage model via the use of a fertility component. Finally methods to integrate linguistic annotation into the input and output sentences, respectively, are discussed, followed by the possibility of linguistically structured NMT models. Koehn points out that these challenges require future work, but could again be a profitable area of research for newcomers to the field looking to make a name for themselves.

Chapter 16 reviews current challenges in NMT and compares how well NMT models perform compared to SMT systems. The first comparison that is reviewed is domain mismatch. For the review, five different NMT and SMT systems were trained and an additional system was trained on all five domains. The resulting performance is given for the NMT and SMT systems. On in-domain data the performance is similar, but for out-of-domain data the performance of the NMT systems is much worse, largely due to the amount of training data available. In this section we see how translation performance is related to increasing amounts of data. Unsurprisingly, phrase-based SMT models (Koehn et al. 2003) are more effective on small amounts of data, compared to NMT models that are more effective on large amounts

of data. When reviewing the handling of rare words, NMT systems perform better on unknown and low frequency words than SMT. The effects of different types of noisy data are also reviewed. This section begins by explaining what real-world noise in data actually is, and then describing how different types of synthetic noise can be created and added to training sets in order to study their impact on NMT and SMT systems. After this the impact on translation quality is reviewed by training models on training sets with different types of noisy data added to them. As far as beam search is concerned, the performance of SMT models always improves as the beam size increases, but this is not always the case for NMT. Finally, the word alignment quality for the attention model used in NMT is compared to alignments obtained from fast-align used in SMT.

In Chapter 17, the final one in the book, the challenge of analysing NNs and visualising and inspecting the internal structure of these networks is addressed. The first challenge is error analysis. Although automatic evaluation metrics serve as good indicators of quality improvement, these metrics do not provide any insight into the type of errors that occur in NMT model translations. Some of the commonly made translation errors by NMT models are listed and described in terms of linguistic error categories. This is followed by a discussion of the findings of a real-world example in which human post-editors corrected MT output sentences to assess the quality of different systems. In the last part of this section, Koehn discusses how challenge sets or synthetic languages can be used as additional methods for testing the performance of NMT systems. In the next section of this chapter, methods for visualising model parameters and node values during inference are discussed. Word embeddings can be projected into a two-dimensional space which can be used to examine whether or not semantically similar words have similar vector representations. Similarly, word senses can be visualised by plotting each encoder state for different occurrences of a specific word, which reveals syntactic clusters. As for SMT, word alignment would be used to determine how important an input word is for the translation of a given output word; in NMT, this can be visualised through the attention weights. Decoding states are more difficult to visualise and Koehn refers the reader to Tran et al. (2016) and the inspection tool presented by Strobelt et al. (2018) to understand how these states can be visualised. Beam search can also be visualised using a toolkit such as that of Lee et al. (2017). The final two sections focus on identifying individual neurons in a given layer and tracing decisions back to inputs. Two examples are given on inspecting individual neurons, which show that relevant information for certain properties is stored in individual neurons. Koehn goes on to explain why it is necessary to trace decisions back to inputs during error analysis and discusses two methods can be used for this purpose, namely layer-wise relevance propagation (Ding et al. 2017) and saliency (Ding et al. 2019).

4 Conclusion

In summary, we believe that this book lays a great foundation for the understanding of NMT. Combined with additional diagrams and figures, it is easy to comprehend the various concepts in relation to the NMT techniques discussed. The further

readings provided at the end of most chapters serve as a useful resource for current topics in the field and more detail of each topic presented in the various chapters. This book can essentially be viewed as an important contribution to the increasingly important area of neural MT, which will be a great help to NLP researchers, scientists, academics, undergraduate or postgraduate students, and MT researchers and users in particular.

Acknowledgements This work was part-funded by Science Foundation Ireland through the SFI Centre for Research Training in Machine Learning (18/CRT/6183). The ADAPT Centre for Digital Content Technology is funded under the SFI Research Centres Programme (Grant 13/RC/2106), and is co-funded under the European Regional Development Fund.

Funding Open Access funding provided by the IReL Consortium.

References

Banerjee S, Lavie A (2005) METEOR: an automatic metric for MT evaluation with improved correlation with human judgments. In: Proceedings of the ACL workshop on intrinsic and extrinsic evaluation measures for machine translation and/or summarization, Ann Arbor, MI, pp 65–72

Carl M, Way A (Eds) (2003) Recent advances in example-based machine translation. Kluwer Academic Publishers, Dordrecht, The Netherlands

Choi H, Cho K, Bengio Y (2018) Fine-grained attention mechanism for neural machine translation. Neurocomputing 284:171–176

Chung J, Cho K, Bengio Y (2016) A character-level decoder without explicit segmentation for neural machine translation. In: Proceedings of the 54th annual meeting of the association for computational linguistics, (Vol 1: Long Papers), Berlin, Germany, pp 1693–1703

Devlin J, Chang M-W, Lee K, Toutanova K (2019) BERT: pre-training of deep bidirectional transformers for language understanding. In: Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics: human language technologies, vol 1 (Long and Short Papers), Minneapolis, MN, pp 4171–4186

Ding Y, Liu Y, Luan H, Sun M (2017) Visualizing and understanding neural machine translation. In: Proceedings of the 55th annual meeting of the association for computational linguistics (Volume 1: long papers), Vancouver, BC, pp 1150–1159

Ding S, Xu H, Koehn P (2019) Saliency-driven word alignment interpretation for neural machine translation. In: Proceedings of the fourth conference on machine translation (Volume 1: research papers), Florence, Italy, pp 1–12

Farajian MA, Turchi M, Negri M, Federico M (2017) Multi-domain neural machine translation through unsupervised adaptation. In: Proceedings of the second conference on machine translation, Copenhagen, Denmark, pp 127–137

Gal Y, Ghahramani Z (2016) A theoretically grounded application of dropout in recurrent neural networks. In: Proceedings of the 30th international conference on neural information processing systems, Barcelona, Spain, pp 1027–1035

He D, Xia Y, Qin T, Wang L, Yu N, Tie-Yan Lu, Wei-Ying M (2016) Dual learning for machine translation. In: Proceedings of the 30th international conference on neural information processing systems, Barcelona, Spain, pp 820–828

Kalchbrenner N, Grefenstette E, Blunsom P (2014) A convolutional neural network for modelling sentences. In: Proceedings of the 52nd annual meeting of the association for computational linguistics (Volume 1: long papers), Baltimore, MD, pp 655–665

Koehn P (2009) Statistical machine translation. Cambridge University Press, Cambridge, UK

Koehn P, Och FJ, Marcu D (2003) Statistical phrase-based translation. In: HLT-NAACL 2003: conference combining Human Language Technology conference series and the North American Chapter of the Association for Computational Linguistics conference series, Edmonton, AB, pp 48–54

Kudo T, Richardson J (2018) Sentencepiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. In: Proceedings of the 2018 conference on empirical methods in natural language processing: system demonstrations, Brussels, Belgium, pp 66–71

Lee J, Shin J-H, Kim J-S (2017) Interactive visualization and manipulation of attention-based neural machine translation. In: Proceedings of the 2017 conference on empirical methods in natural language processing: system demonstrations, Copenhagen, Denmark, pp 121–126

Ling W, Trancoso I, Dyer C, Black AW (2015) Character-based neural machine translation. arXiv:1511.04586

Läubli S, Sennrich R, Volk M (2018) Has machine translation achieved human parity? A case for document-level evaluation. In: Proceedings of the 2018 conference on empirical methods in natural language processing, Brussels, Belgium, pp 4791–4796

Mikolov T, Chen K, Corrado G, Dean J (2013) Efficient estimation of word representations in vector space. arXiv:1301.3781

Papineni K, Roukos S, Ward T, Zhu W-J (2002) BLEU: a method for automatic evaluation of machine translation. In: Proceedings of the 40th annual meeting of the association for computational linguistics, Philadelphia, PA, pp 311–318

Pennington J, Socher R, Manning CD (2014) Glove: global vectors for word representation. In: Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP), Doha, Qatar, pp 1532–1543

Peters M, Neumann M, Iyyer M, Gardner M, Clark C, Lee K, Zettlemoyer L (2018) Deep contextualized word representations. In: Proceedings of the 2018 conference of the North American Chapter of the Association for Computational Linguistics: human language technologies, Volume 1 (long papers), New Orleans, LA, pp 2227–2237

Poncelas A, Shterionov D, Way A, de Buy Wenniger GM, Passban P (2018) Investigating backtranslation in neural machine translation. In: Proceedings of the 21st annual conference of the European Association for Machine Translation (EAMT 2018), Alicante, Spain, pp 249–258

Ruder S, Vulić I, Søgaard A (2019) A survey of cross-lingual word embedding models. J Artif Intell Res 65(1):569–631

Sennrich R, Haddow B, Birch A (2016a) 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), Berlin, Germany, pp 1715–1725

Sennrich R, Haddow B, Birch A (2016b) 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), Berlin, Germany, pp 86–96

Snover M, Dorr B, Schwartz R, Micciulla L, Makhoul J (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, Austin, TX, pp 223–231

Sri Ram Kothur S, Knowles R, Koehn P (2018) Document-level adaptation for neural machine translation. In: Proceedings of the 2nd workshop on neural machine translation and generation, Melbourne, Australia, pp 64–73

Strobelt H, Gehrmann S, Behrisch M, Perer A, Pfister H, Rush AM (2018) Seq2seq-vis: A visual debugging tool for sequence-to-sequence models. IEEE Trans Visual Comp Graphics 25(1):353–363

Toral A, Castilho S, Hu K, Way A (2018) Attaining the unattainable? Reassessing claims of human parity in neural machine translation. In: Proceedings of the third conference on machine translation: research papers, Brussels, Belgium, pp 113–123

Tran K, Bisazza A, Monz C (2016) Recurrent memory networks for language modeling. In: Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, San Diego, CA, pp 321–331,

Tu Z, Lu Z, Liu Y, Liu X, Li H (2016) Modeling coverage for neural machine translation. In: Proceedings of the 54th annual meeting of the association for computational linguistics (Volume 1: long papers), Berlin, Germany, pp 76–85

van der Wees M, Bisazza A, Monz C (2017) Dynamic data selection for neural machine translation. In: Proceedings of the 2017 conference on empirical methods in natural language processing, Copenhagen, Denmark, pp 1400–1410

Vaswani A, Shazeer N, Parmar N, Uszkoreit J, Jones L, Gomez AN, Kaiser L, PolosukhinI (2017) Attention is all you need. In: Advances in neural information processing systems, Long Beach, CA, pp 6000–6010

Wang W, Peter JT, Rosendahl H, Ney H (2016) Character: Translation edit rate on character level. In: Proceedings of the first conference on machine translation: Volume 2, shared task papers, Berlin, Germany, pp 505–510

Way A (2018) Quality expectations of machine translation. In: Moorkens J, Castilho S, Gaspari F, Doherty S (Eds). Translation quality assessment: from principles to practice, Springer, Cham, Switzerland, pp 159–178

Zaremba W, Sutskever I, Vinyals O (2014) Recurrent neural network regularization. arXiv:1409.2329

Zhang X, Kumar S, Khayrallah H, Murray K, Gwinnup J, Martindale MJ, McNamee P, Duh K, Carpuat M (2018) An empirical exploration of curriculum learning for neural machine translation. arXiv:1811.00739