Take Help from Elder Brother: Old to Modern English NMT with Phrase Pair Feedback

Sukanta Sen[1], Mohammad Hasanuzzaman[2], Asif Ekbal[1], Pushpak Bhattacharyya[1], and Andy Way[2]

[1] Department of Computer Science & Engineering,
Indian Institute of Technology Patna, India
{sukanta.pcs15, asif, pb}@iitp.ac.in
[2] ADAPT Centre, School of Computing, Dublin City University, Ireland
hasanuzzaman.im@gmail.com, andy.way@adaptcentre.ie

Abstract. Due to the ever-changing nature of the human language and the variations in writing style, age-old texts in one language may be incomprehensible to a modern reader. In order to make these texts familiar to the modern reader, we need to rewrite them manually. But this is not always feasible if the volume of texts is very large. In this paper, we present this rewriting task as a neural machine translation (NMT) problem. We propose an effective approach for training NMT system using a tiny parallel corpus comprising of only 2.7k parallel sentences. We inject parallel phrase pairs extracted using Statistical Machine Translation (SMT) as additional training examples to NMT. We choose publicly available old-modern English parallel texts for our experiments. Evaluation results show that our proposed approach outperforms the baseline NMT system by more than 18 BLEU points without using any additional training data.

Keywords: Neural Machine Translation, Low resource NMT, Phrase Extraction, Old-Modern English.

1 Introduction

Human languages are constantly evolving and changing over time to reflect socio-cultural changes, fit current conventions, mores, expressions, and needs. This change in a language often requires rewriting the old texts for the modern readers in the same language. In line with global trends, old texts are increasingly available in the forms that computer can process. These ever expanding records (e.g. historical records, scanned books, academic papers, large-scale corpora, maps etc.)—either digitally born or reconstructed through digitization pipelines—are too big to be rewritten manually. Humanities researchers including historians have a keen interest in computational approaches to process and study digitized old texts for research, writing, and dissemination of knowledge.

In this paper, we pose this rewriting task as a problem of machine translation and use Neural Machine Translation (NMT) as a framework to solve. The NMT

[3,6,11,16,25] has recently drawn significant attention to the researchers due to its encouraging performance on publicly available benchmark datasets [5] and rapid adoption in production systems [8,15,26]. The key points of NMT are: it generates fluent outputs and it can be implemented as a single end-to-end neural system unlike long-dominant phrase-based SMT [20] which combines many submodules. However, the task is not a trivial one. It requires a huge amount of parallel data (often in the range of millions) for building a good NMT system. In the absence of sufficient amount of data, a model learns poorly because of low-counts of source-target units, and makes NMT suffer from the adequacy problem. This is also true in case of an old-modern parallel texts, where we have a dearth of parallel corpus.

In this work, we propose an NMT system using a very small parallel corpus (approximately 2.7k parallel sentences for old-modern English). We extract phrase pairs from the training data with the help of phrase-based SMT [20] training, and augment the original training corpus by adding these phrase pairs as parallel sentences. By phrase, we do not necessarily mean any linguistic phrase – it is rather a consecutive sequence of words. We evaluate our approach using the BLEU [21], METEOR [4] and TER [24] metrics against the baseline model. The baseline is trained using original training data only. Our experiments show that our proposed approach attains significant performance gain over the baseline model. We also compare our approach with the well established back-translation [23] method under different settings. We summarize the contributions of our current work as follows:

- We propose an effective NMT approach with feedback from SMT phrases for translating old-to-modern English texts.
- We empirically show that when we do not have enough parallel sentences, parallel phrases, extracted from the training data, can bring significant improvement.
- We also find that our approach can further improve a model trained using back-translation based method.

2 Related Works

Archer et al. [1] presented a summary of previously attempted manual and semi-automatic methods to map historical spelling variants to modern equivalents in order to use it for several natural language processing (NLP) applications.

Domingo et al. [9] used SMT to modernize historical documents. To the best of our knowledge, there exists no significant attempt which aims to automatically translate such old English text into the modern English using NMT.

However, our research is not specific to the old-modern English. Our approach is also applicable to the low-resource scenarios. For training an NMT system, we require a large amount of parallel corpus which is not readily available for every languages and domains. In order to address this issue, however, there have been few attempts to build the NMT systems for low-resource language pairs [10,12,23,27] by incorporating huge monolingual corpus in both the source and target sides.

Sennrich et al. [23] have incorporated monolingual data on the target side to investigate two methods of filling the source side of the monolingual data. In the first method, they have used a dummy source sentence for every target sentence and in the second method, they used a synthetic source sentence obtained via back-translation. They claimed that the second method is more effective. However, if there is not enough parallel data, quality of the back-translation is again a problem.

Zhang and Zong [27] explored the effect of incorporating large-scale source-side monolingual data in NMT in different ways. In the first approach, inspired by [23], they first built a baseline system and then obtained parallel synthetic data by translating the monolingual data. This parallel data along with the original data is used for training an attention-based encoder-decoder NMT system. The second method used the multi-task learning framework to generate the target translation

Arthur et al. [2] have proposed a model to incorporate translation lexicons through calculating lexical predictive probability and adding this probability to the input of the softmax. Zoph et al. [28] applied transfer learning for low-resourced NMT. Although there have been attempts to expand the training data through back-translated monolingual corpus, the effect of adding source-target phrases into the training data is less explored. We hypothesize that augmenting phrase pairs into training data may be useful for generating adequate translations.

3 Proposed Method

We focus on a low-resource scenario where we have a very small parallel data only. To deal with this situation, we add the phrase pairs extracted from the training corpus as feedback to the NMT framework during training. The proposed method uses a state-of-the-art attention-based encoder-decoder NMT architecture [3]. Here, we briefly describe the architecture first and then present the details of the proposed method.

3.1 Overview of NMT

The goal of NMT is to translate a sequence of source words into a sequence of target words with the help of a large neural network. The basic architecture of an NMT uses two recurrent neural networks, one is called encoder and other is known as decoder. The encoder converts the source sentence into a dense fixed-length vector and then the decoder generates target sentence from that vector. But the main drawback of this encoder-decoder approach is that it fails drastically as length of the input sentence grows. The encoder-decoder approach assumes that the encoder can encode the whole sentence into a fixed length vector, which is not realistic, specifically for the longer sentences. To mitigate this drawback, Bahdanau et al. [3] came up with an idea which focused on the whole input sentence while generating the outputs.

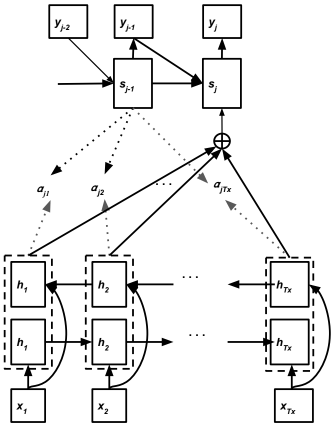

Fig. 1: Attention-based encoder-decoder architecture

Formally, given a sequence of source words x $(= x_1, x_2, x_3, ..., x_n)$ and the previously translated $i-1$ words y $(= y_1, y_2, y_3, ..., y_{i-1})$, the probability of the ith translated word y_i is calculated as:

$$p(y_i|s_{i-1}, y_{i-1}, c_i) = softmax(W_o t_i) \tag{1}$$

where t_i, the input to the $softmax$ is computed as:

$$t_i = tanh(W_s s_i + W_e y_{i-1} + W_c c_i) \tag{2}$$

where W_s, W_e, W_c, W_o are the model parameters. The hidden state s_i in the decoder at time step i is computed as:

$$s_i = g(s_{i-1}, y_{i-1}, c_i) \tag{3}$$

Here g is a nonlinear transformation function, which is usually a Long short-term memory (LSTM) [14] or a gated recurrent unit (GRU) [6], and c_i is the context vector at time step i, which is calculated as a weighted sum of the input annotations h_j:

$$c_i = \sum_{j=1}^{Tx} \alpha_{ij} \mathrm{h}_j \tag{4}$$

where Tx is the length of the source sequence and h_j is the encoder hidden state at jth time step and computed using a nonlinear transformation function as:

$$h_j = f(h_{j-1}, x_j) \tag{5}$$

The normalized weight α_{ij} for h_j is calculated as:

$$\alpha_{ij} = \frac{\exp(e_{ij})}{\sum_{k=1}^{Tx} \exp(e_{ik})} \tag{6}$$

$$e_{ij} = V_a^T tanh(U_a s_{i-1} + W_a h_j) \tag{7}$$

where V_a, U_a and W_a are the trainable parameters. All of the parameters in the NMT model are optimized to maximize the following conditional log-likelihood of the N parallel sentences

$$\ell(\theta) = \frac{1}{N} \sum_{n=1}^{N} \sum_{j=1}^{Ty} \text{log} p(y_j | s_j, y_{j-1}, c_j) \tag{8}$$

where Ty is the length of the target sequence.

3.2 Phrase Augmentation

Our proposed method uses the state-of-the-art attention-based encoder-decoder [3] NMT architecture. Apart from feeding sentence pairs into the attention-based encoder-decoder, we also feed phrase pairs as training examples. This gives an illusion of having a larger corpus. In order to do this, we first extract parallel phrases from the parallel corpus and then add these parallel phrases in the training set. We use the Moses [19] SMT system. We train a source-target phrase-based SMT [20] and extract all the phrase pairs from the phrase table. However, all of these parallel phrases are not sound, i.e. there can be some wrong source-target phrases [20]. We set different conditions while choosing the phrases from the phrase table. Assuming every source phrase e is aligned to a set of target phrase $F = (f_1, f_2, ..., f_n)$, we consider all or some target phrases for a source phrase. Note that n may vary for each source phrase. We select three sets of phrase pairs from the phrase table.

(i) For first set: we select all the parallel phrases from the phrase table.
(ii) For second set: we select phrase pairs (e, f_t) with $P(f_t|e) \geq 0.5$
(iii) For third set: we select phrase pairs (e, f_t) with $P(f_t|e) = 1$

where f_t is a target phrase for a source phrase e. Since the number of phrase pairs is larger than the number of original parallel sentences, to maintain a fair ratio between them, we use the following formula for combining phrase pairs with the original training set.

$$\begin{aligned} Augmented\ Corpus = N \times Original\ corpus \\ + Extracted\ phrase\ pairs \end{aligned} \tag{9}$$

We combine the extracted set (of parallel phrases) with N times of the original corpus, where, N is calculated as

$$N = \frac{Number\ of\ extracted\ phrase\ pairs}{Number\ of\ original\ parallel\ sentences}$$

Otherwise, training set will contain mostly the phrases and since the phrases are smaller in length, they may make the model biased towards the phrase length.

4 Data Sets

We use publicly available old-modern parallel text[3]. As old English texts, we use publicly available *The Homilies of the Anglo-Saxon Church* [4] by Ælfric of Eynsham (c.950 – c.1010) who was a prolific author in old English, and its translation by Benjamin Thorpe (c.1782 – c.1870) as modern English texts. We call it OE-ME corpus. The OE-ME corpus is tiny in size and it has 720 parallel paragraphs divided into 40 sections. Most of the parallel segments have equal number of parallel sentences which help in aligning the parallel sentences. For experiments, we randomly split it into three sets: *train*, *test*, and *dev* containing 2,716, 500, and 500 sentences, respectively. For tokenization, we use *tokenizer.perl* of the Moses SMT system. We observe that OE has a larger vocabulary than ME, however, ME has more tokens than OE in all *train*, *test*, and *dev* sets. This also gives an intuition that even though they belong to the same language, they are not linguistically same at all. We also use the openly available English Bible corpus[5] [7] for back-translation [23] into old English. Since our original training corpus comprises of religious texts, we choose the Bible corpus. It has approximately 31k modern English sentences. Details of the datasets are presented in Table 1.

Table 1: Number of sentences for different data sets

	train	*dev*	*test*	*Bible*
#Sentence	2,716	500	500	31,102

5 Experimental Setup

We use Nematus [22] for training the NMT models. All our NMT models are based on attention-based encoder-decoder approach and trained at word level. We set embedding size as 128, hidden size as 256 and learning rate as 0.001. The models are trained with mini-batch size of 40 and we restrict the maximum sentence length to 80 words. We consider all vocabulary words (vocabulary size for each system has been shown in Table 2). We use the Adam optimizer [17] for

[3] `https://en.wikisource.org/wiki/The_Homilies_of_the_Anglo-Saxon_Church`

[4] `https://en.wikisource.org/wiki/The_Homilies_of_the_Anglo-Saxon_Church`

[5] https://github.com/christos-c/bible-corpus/blob/master/bibles/English.xml

optimizing the models. The training stops on meeting the early-stopping criteria[6]. We save a model on every 2,500 updates. For testing, we set beam size as 3 and select the model that produced the highest BLEU score on the development set for scoring the test set. For other parameters, default values of Nematus were used. We use the Moses [19] for training the phrase-based SMT (*PBSMT*) and as well as for phrase extraction. For training, we keep the following settings in the Moses: grow-diag-final-and heuristics for word alignment, msd-bidirectional-fe for reordering model, and 4-gram language model (LM) with modified Kneser-Ney smoothing [18] using KenLM [13]. However, we note that the order of LM does not affect the phrase table. We train the following different types of NMT systems:

(i) *Baseline-NMT*: The NMT model is trained using only original parallel corpus.

(ii) *BackTrans*: The NMT model is trained using original corpus along with the back-translated parallel sentences (of size 10k, 20k, and 31k). Back-translated corpus (BT) is generated (translating the Bible corpus from modern to old English) using a SMT system trained on the original training corpus.

(iii) *Phrase-Augmented*: These systems are trained to improve the *Baseline-NMT* and *BackTrans* systems by injecting the phrase pairs extracted from the respective training data. We train three types of *Phrase-Augmented* systems (see Table 2):
– *Type-A*: For these systems, the training data is augmented using phrases extracted from original training data only.
– *Type-B*: For these systems, the training data is augmented using phrases extracted from original training data and back-translated data together.
– *Type-C*: For these systems, we add back-translated data with the original data.

6 Results and Analysis

Table 2 summarizes the results of different systems. We can see that SMT is still the best choice in the absence of sufficient parallel corpus. *However, our motivation is not to beat the SMT system, but to develop an appropriate NMT based system in the absence of sufficiently large parallel corpus.* It is well established that SMT performs better in the absence of sufficient amount of corpus. In contrast, NMT is better at fluency and being an end-to-end system, it is easy adoptable and scalable.

From Table 2, it is evident that the baseline NMT system *Baseline-NMT* is outperformed by all of our proposed systems. The third system (B_N+$Phrase_{Org.}$) obtains the best performance among all the three proposed systems (with phrase extracted from original corpus only, i.e. *Type-A*). Though back-translation (B_N+

[6] We use early stopping based on BLEU measure with early-stopping patience value 10. All the models run for 110-140k (approx.) updates before early-stopping.

Table 2: Scores of different systems in terms of BLUE, METEOR and TER. p is the probability of a phrase pair. Org:original training data. $PHR_{Org.}$: phrase pairs extracted from *Org.* BT: Parallel corpus obtained through back-translation. $PHR_{Org+BT,p}$: phrase pairs, having probability p, extracted from *Org.* and BT. **Data Size** column shows the training data size using the Eq. 9. **Vocab** column shows vocabulary sizes for old and modern English

SYSTEM		**Data Size**	**Vocab**		**BLEU**	**METEOR**	**TER**
			old	mod			
	PBSMT	2,716	8,878	5,102	39.95	36.96	37.99
	Baseline-NMT (B_N)	2,716			**10.03**	**15.95**	**90.06**
Back-Trans	B_N+10k BT	12,716	15,067	10,948	21.90	24.95	64.66
	B_N+20k BT	22,716	17,341	13,162	24.23	25.83	58.15
	B_N+BT	33,818	19,083	14,859	**29.10**	**30.70**	**52.48**
Proposed Approach							
Type-A	B_N+$PHR_{Org,p=1.0}$	341,659	8,878	5,102	20.83	25.84	84.66
	B_N+$PHR_{Org,p\geq0.5}$	385,015			25.41	27.79	69.42
	B_N+PHR_{Org}	485,739			**28.76**	**28.56**	**56.37**
Type-B	B_N+$PHR_{Org+BT,p=1.0}$	4,850,270	19,080	14,855	25.30	26.50	63.33
	B_N+$PHR_{Org+BT,p\geq0.5}$	5,094,325			27.93	28.76	60.58
	B_N+$PHR_{Org.+BT}$	6,068,402			25.17	26.95	68.76
Type-C	B_N+BT+PHR_{Org}	512,613	19,083	14,859	**32.35**	**31.88**	**50.36**
	B_N+BT+PHR_{Org+BT}	6,073,265			**29.37**	**30.31**	**56.02**

BT) performs better than our third system B_N+$Phrase_{Org.}$ by a very small (0.34 BLEU) margin, it is to be noted that it requires significantly large amount of external data compared to the original training data. Our proposed system (B_N+$Phrase_{Org.}$), however, outperforms B_N+10k BT and B_N+20k BT by 6.86 and 4.53 BLEU points, respectively. The improvements are also consistent with respect to the other evaluation metrics METEOR and TER as well. Evaluation suggests that our proposed model can provide a promising solution when we do not have a sufficient amount of parallel corpus and a large amount of monolingual corpus. We also note that when we select phrase pairs with probabilities $p \geq 0.5$ and $p = 1$ (for *Type-A*) we lose many phrase pairs (with probabilities $p \leq 0.5$ and $p < 1$, respectively). We can see that considering all phrases extracted from original training data results the best performance (B_N+$Phrase_{Org.}$). However, when we consider all phrases, there will be some wrong parallel phrases as well but the result shows that they do not affect the overall BLEU score.

We also experiment using phrases extracted from the original training data *Org* and back translated (BT) data combined. We observe performance improvement when these are used for training the model (c.f. performance of B_N+ $PHR_{Org+BT,p=1.0}$ and $B_N + PHR_{Org+BT,p\geq0.5}$ improve over B_N+ $PHR_{Org,p=1.0}$ and $B_N + PHR_{Org,p\geq0.5}$, respectively). However, B_N+ PHR_{Org+BT} does not improve compared to B_N+ PHR_{Org} because of the following reason: here we consider all the phrases and a significant number of phrases are wrong as back-translated parallel corpus has many wrong source sentences. Our experiments (*Type-C*) show that *BackTrans* can be further improved by augmenting phrases. We note that systems B_N+PHR_{Org+BT} and B_N+BT+PHR_{Org+BT} perform poorer than their preceding counterparts, i.e. systems B_N+PHR_{Org} and B_N+BT+PHR_{Org}, respectively. This might have happened due to a number of incorrect phrases that are being added to original training set as a result of phrase extraction from *Org+BT*. This approach does not generate perfect phrases because BT itself contains many wrong source sentences.

Table 3: Output examples of different NMT systems. *Output 1* is from B_N+ $Phrase_{Org}$ and *Output 2* is from B_N+ BT+$Phrase_{Org}$

Old → ***Modern***	
Source	Uton nu gehyran be õan Halgan Gaste, hwæt he sý.
Reference	Let us now hear concerning the Holy Ghost, what he is.
Baseline-NMT	Let now say as much as a Holy Ghost speak Blessed much a Holy Ghost speak, let the Holy Ghost how he might clear himself , what he might manifestly promised us.
Output 1	let us now be prophesied of the Holy Ghost , what he might not .
Output 2	let us now hear concerning the Holy Ghost , what he might overcome them .

To conclude the analysis, we observe that when we add phrases from the original training data only, we can select all the phrases. However, when we extract phrases from the training and back-translated data combined, we need to ignore many phrases because back-translated data produces many wrong phrases.

Table 3 shows few outputs produced by the proposed systems. We observe that the output of the baseline system is not adequate and many parts of the output are repetitive. For example the phrases "*Holy Ghost*" and "*he might*" are output multiple times. This is because the baseline model (trained on only original training set) does not learn the mapping between theses phrases as the corpus is very small. In contrast, our proposed system generates better translation.

7 Conclusion

In this paper, we proposed a method to train an NMT system that performs very effectively on a tiny parallel corpus. We extracted phrase pairs from the original training corpus and used them to augment the training data. This gives an illusion of having more training examples. We choose publicly available old-modern English parallel corpus which comprises only 2.7k sentences, and posed the rewriting task of old English to modern English as NMT task. We experimentally showed that our approach can significantly improve an NMT system when have we very little amount of training data. We followed standard attention-based encoder-decoder network and our approach improved the baseline old-to-modern English NMT system by a margin of up to 18 BLEU points. Back-translation has been shown to improve the baseline significantly in many previous NMT tasks. However, quality of back-translation highly depends on the size of original training data. We showed that in a extremely low-resource scenario like old-modern English translation, our approach further improves an NMT system build on top of back-translation.

Our approach augments the original training set and is not specific to any NMT architecture. Thus it can be used in any NMT setting. In this work, we used attention-based encoder-decoder architecture for a specific problem of old-to-modern English translation. In future, we would like apply our approach to various NMT architectures for many low-resource language pairs. In addition to that, we would like to see if the extracted phrases improve an NMT system when the original training corpus is sufficiently large.

8 Acknowledgments

Asif Ekbal acknowledges Young Faculty Research Fellowship (YFRF), supported by Visvesvaraya PhD scheme for Electronics and IT, Ministry of Electronics and Information Technology (MeitY), Government of India, being implemented by Digital India Corporation (formerly Media Lab Asia). Mohammed Hasanuzzaman and Andy Way would like to acknowledge ADAPT Centre for Digital Content Technology, funded under the SFI Research Centres Programme (Grant 13/RC/2106) and is co-funded under the European Regional Development Fund.

References

1. Archer, D., Kytö, M., Baron, A., Rayson, P.: Guidelines for normalising early modern english corpora: Decisions and justifications. ICAME Journal **39**(1), 5–24 (2015)
2. Arthur, P., Neubig, G., Nakamura, S.: Incorporating discrete translation lexicons into neural machine translation. In: Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing. pp. 1557–1567 (2016)
3. Bahdanau, D., Cho, K., Bengio, Y.: Neural machine translation by jointly learning to align and translate. International Conference on Learning Representation(ICLR) (2015)
4. Banerjee, S., Lavie, A.: 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. pp. 65–72. Association for Computational Linguistics (2005)
5. Bojar, O., Chatterjee, R., Federmann, C., Graham, Y., Haddow, B., Huck, M., Yepes, A.J., Koehn, P., Logacheva, V., Monz, C., et al.: Findings of the 2016 conference on machine translation. In: ACL 2016 First Conference on Machine Translation (WMT16). pp. 131–198. The Association for Computational Linguistics (2016)

6. Cho, K., Van Merriënboer, B., Bahdanau, D., Bengio, Y.: On the properties of neural machine translation: Encoder-decoder approaches. In: Proceedings of SSST-8, Eighth Workshop on Syntax, Semantics and Structure in Statistical Translation. pp. 103–111 (2014)
7. Christodouloupoulos, C., Steedman, M.: A massively parallel corpus: the bible in 100 languages. Language resources and evaluation **49**(2), 375–395 (2015)
8. Crego, J., Kim, J., Klein, G., Rebollo, A., Yang, K., Senellart, J., Akhanov, E., Brunelle, P., Coquard, A., Deng, Y., et al.: Systran's pure neural machine translation systems. arXiv preprint arXiv:1610.05540 (2016)
9. Domingo, M., Chinea-Rios, M., Casacuberta, F.: Historical documents modernization. The Prague Bulletin of Mathematical Linguistics **108**(1), 295–306 (2017)
10. Fadaee, M., Bisazza, A., Monz, C.: Data augmentation for low-resource neural machine translation. In: Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers). pp. 567–573. Association for Computational Linguistics, Vancouver, Canada (July 2017)
11. Forcada, M.L., Ñeco, R.P.: Recursive hetero-associative memories for translation. In: International Work-Conference on Artificial Neural Networks. pp. 453–462. Springer (1997)
12. Gulcehre, C., Firat, O., Xu, K., Cho, K., Bengio, Y.: On integrating a language model into neural machine translation. Computer Speech & Language **45**, 137–148 (2017)
13. Heafield, K.: Kenlm: Faster and smaller language model queries. In: Proceedings of the Sixth Workshop on Statistical Machine Translation. pp. 187–197. Association for Computational Linguistics (2011)
14. Hochreiter, S., Schmidhuber, J.: Long short-term memory. Neural computation **9**(8), 1735–1780 (1997)
15. Junczys-Dowmunt, M., Dwojak, T., Hoang, H.: Is neural machine translation ready for deployment? a case study on 30 translation directions. In: In Proceedings of the International Workshop on Spoken Language Translation (IWSLT) (2016)
16. Kalchbrenner, N., Blunsom, P.: Recurrent continuous translation models. In: Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing. pp. 1700–1709 (2013)
17. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. International Conference on Learning Representation(ICLR) (2015)
18. Kneser, R., Ney, H.: Improved backing-off for m-gram language modeling. In: Acoustics, Speech, and Signal Processing, 1995. ICASSP-95., 1995 International Conference on. vol. 1, pp. 181–184. IEEE (1995)
19. Koehn, P., Hoang, H., Birch, A., Callison-Burch, C., Federico, M., Bertoldi, N., Cowan, B., Shen, W., Moran, C., Zens, R., et al.: Moses: Open source toolkit for statistical machine translation. In: Proceedings of the 45th annual meeting of the ACL on interactive poster and demonstration sessions. pp. 177–180. Association for Computational Linguistics (2007)
20. Koehn, P., Och, F.J., Marcu, D.: Statistical phrase-based translation. In: Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology-Volume 1. pp. 48–54. Association for Computational Linguistics (2003)
21. Papineni, K., Roukos, S., Ward, T., Zhu, W.J.: BLEU: a Method for Automatic Evaluation of Machine Translation. In: Proceedings of the 40th annual meeting on association for computational linguistics. pp. 311–318. Philadelphia, Pennsylvania (2002)

22. Sennrich, R., Firat, O., Cho, K., Birch, A., Haddow, B., Hitschler, J., Junczys-Dowmunt, M., Läubli, S., Barone, A.V.M., Mokry, J., et al.: Nematus: a toolkit for neural machine translation. arXiv preprint arXiv:1703.04357 (2017)
23. Sennrich, R., Haddow, B., Birch, A.: Improving neural machine translation models with monolingual data. In: Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, ACL 2016, August 7-12, 2016, Berlin, Germany (2016)
24. Snover, M., Dorr, B., Schwartz, R., Micciulla, L., Makhoul, J.: A study of translation edit rate with targeted human annotation. In: Proceedings of association for machine translation in the Americas. vol. 200 (2006)
25. Sutskever, I., Vinyals, O., Le, Q.V.: Sequence to sequence learning with neural networks. In: Advances in neural information processing systems. pp. 3104–3112 (2014)
26. Wu, Y., Schuster, M., Chen, Z., Le, Q.V., Norouzi, M., Macherey, W., Krikun, M., Cao, Y., Gao, Q., Macherey, K., Klingner, J., Shah, A., Johnson, M., Liu, X., ukasz Kaiser, Gouws, S., Kato, Y., Kudo, T., Kazawa, H., Stevens, K., Kurian, G., Patil, N., Wang, W., Young, C., Smith, J., Riesa, J., Rudnick, A., Vinyals, O., Corrado, G., Hughes, M., Dean, J.: Google's neural machine translation system: Bridging the gap between human and machine translation. CoRR **abs/1609.08144** (2016), `http://arxiv.org/abs/1609.08144`
27. Zhang, J., Zong, C.: Exploiting source-side monolingual data in neural machine translation. In: Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing. pp. 1535–1545 (2016)
28. Zoph, B., Yuret, D., May, J., Knight, K.: Transfer learning for low-resource neural machine translation. In: Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing. pp. 1568–1575. Association for Computational Linguistics, Austin, Texas (2016)