Johns Hopkins University Submission for WMT News Translation Task

Kelly Marchisio
Center for Language and Speech Processing
Johns Hopkins University
kmarchi1@jhu.edu

Yash Kumar Lal
Department of Computer Science
Johns Hopkins University
yash@jhu.edu

Philipp Koehn
Center for Language and Speech Processing
Johns Hopkins University
phi@jhu.edu

Abstract

We describe the work of Johns Hopkins University for the shared task of news translation organized by the Fourth Conference on Machine Translation (2019). We submitted systems for both directions of the English-German language pair. The systems combine multiple techniques – sampling, filtering, iterative backtranslation, and continued training – previously used to improve performance of neural machine translation models. At submission time, we achieve a BLEU score of 38.1 for De-En and 42.5 for En-De translation directions on newstest2019. Post-submission, the score is 38.4 for De-En and 42.8 for En-De. Various experiments conducted in the process are also described.

1 Introduction

This paper describes the Johns Hopkins University (JHU) submission to the Fourth Conference on Machine Translation (WMT19) news translation shared task (Bojar et al., 2019). We built systems for both German-English and English-German. Our attempts are based on previous year's submissions by Edinburgh (model architectures) (Sennrich et al., 2017), Microsoft (data filtering) (Junczys-Dowmunt, 2018), Facebook (backtranslation using sampling) (Edunov et al., 2018), and JHU (continued training on previous years' test sets) (Koehn et al., 2018).

Our models leverage several techniques popular in neural machine translation – backtranslation, continued training (Luong and Manning, 2015) and sentence filtering. We use Transformer-big (Vaswani et al., 2017) models trained on available bitext to generate backtranslations via sampling. These backtranslations are then scored and filtered using dual conditional cross-entropy and cross-entropy difference scores, then added to upsampled bitext (x2). ParaCrawl[1] and Common Crawl[2] are filtered similarly, and added to form the training set for the final models. We refine each final model by performing continued training on the test sets of previous years of WMT. We then perform ensemble decoding using multiple models for each language. Finally, translations are reranked using separately-trained models to obtain the final output. In the De-En direction, scores from a language model also contribute to reranking. In the automatic evaluation, we scored 38.1 on De-En and 42.5 on En-De at submission time. Post-submission, we ensembled more similar models and scored 38.4 on De-En and 42.8 on En-De.

We built our systems using the Marian and Fairseq toolkits.

1.1 Marian

Marian[3] (Junczys-Dowmunt et al., 2018) is a purely C++11 toolkit that allows for creation and training of neural machine translation models efficiently. Most of our models were built using Marian and the sample scripts therein.

1.2 Fairseq

Fairseq[4] (Ott et al., 2019) is a sequence-to-sequence learning toolkit created with a focus on neural machine translation. It contains implementations for various standard NMT architectures and system components. Using this toolkit allows us to use sampling as a method for inference (Edunov et al., 2018).

[1] https://ParaCrawl.eu/index.html
[2] http://CommonCrawl.org
[3] https://marian-nmt.github.io/
[4] https://github.com/pytorch/fairseq

Proceedings of the Fourth Conference on Machine Translation (WMT), Volume 2: Shared Task Papers (Day 1) pages 287–293

2 Motivation

Our work was motivated by three submissions to the news translation task at WMT18. Namely, we combined critical parts of Junczys-Dowmunt (2018), Edunov et al. (2018) and Koehn et al. (2018), and iterated upon them to create our system. Junczys-Dowmunt (2018) was based off of Edinburgh's WMT17 submission (Sennrich et al., 2017).

Our contributions are using filtered backtranslation data and performing hyperparameter search to improve BLEU score gain when performing continued training using previous years' test sets. Models were slightly different for the En-De and De-En directions, which is noted in the subsequent sections.

3 Model Description

Our reproduction of Junczys-Dowmunt (2018), follows the example at `https://github.com/marian-nmt/marian-examples/tree/master/wmt2017-transformer`, using the same data and similar preprocessing. The data is the parallel training bitext provided in the WMT17 shared task, excluding Rapid. Punctuation normalization, tokenization, corpus cleaning and truecasing was applied using Moses (Koehn et al., 2007). The truecaser applied to the clean bitext was trained over the punctuation normalized, tokenized, and cleaned bitext, whereas the truecaser applied to other data, such as the data to backtranslate, was trained on ParaCrawl. We deviated slightly from the example and applied a joint byte pair encoding (BPE) (Sennrich et al., 2016) model that was trained previously over the ParaCrawl German-English bitext to form 32,000 subword units. For the 10 million lines of German monolingual news data to backtranslate, any sentences longer than 100 tokens as well as pairs with source/target length ratio exceeding 9 were discarded after BPE was applied using Moses' `clean-corpus-n.perl`.

Just as Junczys-Dowmunt (2018) replicated Edinburgh's WMT17 results for En-De and upgraded to using the Transformer, we have replicated Junczys-Dowmunt (2018)'s replication with the Transformer-base model. The models were trained on upsampled WMT17 bitext (x2) plus 10M lines of backtranslated German monolingual data. The vocabulary was a joint vocabulary created from the WMT17 bitext and contained 36000 subword units.

Our models for the replication of Junczys-Dowmunt (2018) were trained on a single GPU. For Transformer-base models, we added –maxi-batch-sort src[5]. We additionally added an optimizer delay of 4, and changed the beam size to 6 and the –normalize hyperparameter to 0.6[6]. We trained our Transformer-base models until convergence with early stopping, which was implemented based on Marian word-wise normalized cross-entropy with a patience of 5 and validation occurring every 5000 steps. The maximum training epochs was set to 10. Inference was done using the model with best BLEU score during training.

Model	BLEU
Microsoft Transformer-base (x1)	28.8
+Ensemble	29.4
Our Transformer-base (x1)	29.5
+Ensemble	30.2

Table 1: Reproduction of Microsoft's replication of the University of Edinburgh's submission to WMT17, using the Transformer-base model. Scores are reported on newstest2017. Our single model performance ranged from 28.3-28.6.

Next, we filtered the ParaCrawl data by removing sentence pairs that scored below $e^{-}4$ based on dual conditional cross-entropy filtering, then kept the top 8 million based on cross-entropy difference filtering[7] (Junczys-Dowmunt, 2018). This model's vocabulary included the WMT17 bitext and backtranslated data. The WMT17 bitext was also cleaned after BPE was applied for this model. It achieved a BLEU score of 30.6 on newstest2017, as evidence of the benefit of adding filtered ParaCrawl data.

We also replicated the backtranslation model from Facebook's WMT18 submission in order to use inference by sampling. We first preprocess data in the manner described by Edunov et al. (2018) and then train a Transformer-big model for backtranslation using all available bitext. We used the same hyperparameters mentioned in the original work. The learning rate was set to 0.0001, which is suitable for large batches.

[5] `https://github.com/marian-nmt/marian-dev/issues/184`

[6] Marcin Junczys-Dowmunt, personal communication

[7] Marcin Junczys-Dowmunt, personal communication

All models for the replication of Facebook's submission were trained on a single GPU, which makes it difficult to match results achieved on a large number of GPUs. Fairseq has a training flag to simulate training on multiple GPUs (update-freq) which accumulates updates for a certain number of batches and applies them all at once. Here, the flag was set to 16 (even though it does not replicate the exact settings of the original work). Table 2 shows BLEU scores on newstest2017 for our replication of Facebook AI Research's (FAIR) submission last year.

	Train Set	FAIR '18	Replication
En-De	Bitext	29.5	27.0
	Bitext+top10	32.1	29.6
De-En	Bitext	-	27.8
	Bitext+top10	-	30.6

Table 2: FAIR 2018 Replication

The discrepancy may be due to different batching in the original work and our replication, as the Transformer-big is very sensitive to batch sizes and updates. Edunov et al. (2018) used word batching that we could not match due to memory shortage in the machines we were using. It is likely that this difference in batch size and the distributed versus single-machine training can explain the discrepancies in the numbers. For ideal sampling, we desire a model with as high a BLEU score as possible when translating using beam search, and simultaneously as low a BLEU score as possible when translating using sampling[8].

4 System Components

Our basic training architecture was based off Junczys-Dowmunt (2018), which itself was based of Sennrich et al. (2017).

4.1 Transformer architectures

Using Fairseq, a Transformer-big model was trained over all processed bitext. It was used to translate the prepared monolingual data, employing top-10 sampling (Edunov et al., 2018). Typically, beam search is used to create backtranslated data. Sampling from the model's distribution to create this data allows more room for diverse examples to be generated. Edunov et al. (2018) argue that synthetic data created using this technique sends a "stronger training signal than data generated by beam or greedy search".

Top-10 sampling creates effective, noisy samples and it takes far less time to translate the entire monolingual set than unrestricted sampling.

4.2 Filtering Methods

We applied dual conditional cross-entropy filtering (Junczys-Dowmunt, 2018) and cross-entropy difference filtering (Moore and Lewis, 2010; Junczys-Dowmunt, 2018) to filter our backtranslated data, ParaCrawl, and Common Crawl. ParaCrawl and Common Crawl were combined into a single corpus before filtering.

For both the backtranslation data as well as ParaCrawl and Common Crawl, we first sorted each corpus by "adequacy score", which corresponds to dual conditional cross-entropy filtering. We then removed the lowest-scoring sentences[9], corresponding to an adequecy score threshold of approximately e^{-5} for the backtranslated data, and e^{-4} for ParaCrawl and Common Crawl. Next, we sorted by "domain score", which corresponds to cross-entropy difference filtering, and kept the top 60% of data backtranslated from German, and the top 80% of data backtranslated from English. For ParaCrawl and Common Crawl, we kept the top 50% of data. This data was domain-scored for the target domain. Thus, when the data would be used to train an En-De model, the domain scores were based on cross-entropy difference filtering using models trained with German data, vice-versa for De-En.

Translation models used in dual conditional cross-entropy filtering were shallow RNNs trained on a 1 million line random sample of all available constrained bitext for 2019, excluding ParaCrawl and Common Crawl. The "in-domain" language model for cross-entropy difference filtering was trained on a 1 million line random sample of monolingual News crawl data from WMT16-18, and the "out-of-domain" model was trained on a random 1 million lines from the concatenation of ParaCrawl and Common Crawl.

We discovered a small error in our in-domain language models for cross-entropy difference filtering after submission whereby we had unintentionally filtered out many WMT18 German-side monolingual sentences before creating the language models (LMs). These LMs were used to

[8]Sergey Edunov, personal communication

[9]Marcin Junczys-Dowmunt, personal communication

score both backtranslation as well as ParaCrawl and Common Crawl data.

In total, the filtering methods above resulted in:

- 10.3M lines of ParaCrawl + Common Crawl
- 20.1M lines backtranslated from German
- 13.7M lines backtranslated from English

The filtered data (backtranslations, ParaCrawl, and Common Crawl) was concatenated with 2x upsampled bitext. This results in a final dataset of 40.3M for En-De and 33.9M for De-En. Multiple Transformer-base models were trained over this data using Marian to serve as the primary translation models. A similar method was used to create training data for reranking models, except for these, we reused models whose backtranslations had been generated using beam search. The filtering methods described above resulted in slightly smaller subsets of backtranslated German and English data for the reranking models. Furthermore, the training set for the De-En reranking models was generated by exploiting iterative backtranslation (Hoang et al., 2018; Koehn et al., 2018) along with the filtering methods described. The adequacy score threshold used to filter backtranslations generated via beam search was e^{-4}.

4.3 Continued Training

We fine-tuned the models on newstest2015-18, which closely mirrors the data in the test set. Due to continued training, our models gained up to 1 BLEU point for De-En and up to 1.5 BLEU points for En-De. Multiple such models were then ensembled to perform translations.

5 Training Setup

For our submissions to WMT19, we use similar preprocessing techniques as described for the reproduction of Junczys-Dowmunt (2018), but this time using WMT19 bitext. As a result, 5.2M sentences were obtained. For our submission, we apply Moses' `clean-corpus-n.perl` to the bitext before use.

For backtranslation, we ran a similar preprocessing method on WMT18 News crawl monolingual data. Any sentences with greater than 100 BPE tokens were discarded, leaving us with 34M German monolingual and 24M English monolingual sentences.

Similar to (Sennrich et al., 2017) and (Junczys-Dowmunt, 2018), our training regimen can be divided into these steps:

- Train a Transformer-big model for backtranslation with Fairseq using the clean bitext.
- Backtranslate monolingual data from WMT18 using top-10 sampling.
- Filter backtranslations using domain and adequacy scores.
- Use backtranslated data, upsampled bitext, and filtered ParaCrawl + Common Crawl to train Transformer-base translation models.
- Perform continued training.
- Ensemble decode using translation models.
- Rerank translations using Transformer-base translation models for both language directions, and a language model for De-En.

Reranking models were trained similar to Junczys-Dowmunt (2018) and Sennrich et al. (2017). Our training recipe is as follows:

- Train a shallow RNN model with Marian for backtranslation using clean bitext
- Backtranslate News crawl monolingual data from WMT18 using beam search
- Filter backtranslations using domain and adequacy scores.
- Use backtranslated data, upsampled bitext, and filtered ParaCrawl + Common Crawl to train Transformer-base reranking models.
- Perform continued training.

Since we reused previously-trained models for reranking, the De-En reranking models had additionally undergone filtered iterative backtranslation. The secondary model for backtranslation was a Transformer-base model in the En-De language direction, trained on the upsampled bitext plus the filtered WMT18 News crawl backtranslation data produced by the shallow RNN in the De-En direction. Backtranslations were produced using beam search by the secondary model, concatenated with 2x the clean bitext and the filtered ParaCrawl + Common Crawl, and used to train Transformer-base De-En reranking models.

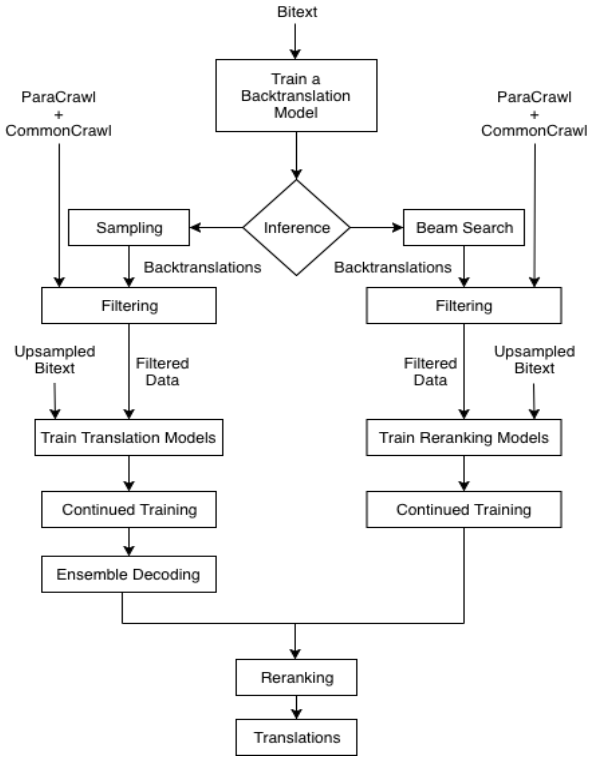

Figure 1: System Architecture. For peculiarities in models of each direction, see Sections 5.1 and 5.2.

BLEU (Papineni et al., 2002) was calculated using the multi-bleu-detok.perl script in Moses.

An overview of our architecture can be found in Figure 1. In the figure, filtered data is comprised of filtered backtranslations, and filtered ParaCrawl and Common Crawl data. All models were trained on a single NVIDIA GeForce GTX 1080Ti GPU.

5.1 English→German

Following the training regimen described above, we first train a Transformer-big model over the original bitext. Hyperparameters used here are the same as the ones used when replicating FAIR. This is used to perform backtranslation of monolingual German data via sampling. The generated data was filtered to the top 60% using both domain and adequacy scoring as described in Section 4.2, before being concatenated with twice the bitext and the filtered ParaCrawl and Common Crawl. Finally, this is used to train two Transformer-base models which are continued trained. We run continued training for 5 epochs at an increased learning rate of 0.001, without the use of a learning rate scheduler. These models are ensembled and used to generate translations which are finally reranked by the reranking models.

For reranking, we replicate the same models mentioned above, except that backtranslations are generated using standard beam search. We retain the same percentage of the backtranslated data. Four such models are created and undergo continued training as described above.

For this direction of the language pair, we corrected the quotation marks of the German translations in a post-processing step.

5.2 German→English

Translation and reranking models for this direction of the language pair were trained the similarly as En-De. We retain the top 80% of the backtranslations by domain score as described in Section 4.2; the ones generating using sampling are used to train the primary translation models, whereas the ones generatd by beam search are used to train the reranking models. We train three Transformer-base translation models that we adapt to previous years' test sets. They run for 5 epochs at an increased learning rate of 0.0005, without the use of a learning rate scheduler. These models are then ensembled to produce a 12-best list of translations.

For reranking in this language direction, we trained our reranking models using iterative backtranslation. We first trained a De-En backtranslation model and used beam search to generate backtranslations for monolingual data from WMT18. The filtered backtranslation data was used along with upsampled bitext to train a second-round En-De backtranslation model. Beam search backtranslations generated using this model, along with clean bitext, ParaCrawl and Common Crawl was used to train the final reranking models. Three of these models were used as the reranking models in conjuction with the three primary models mentioned earlier.

A Transformer-base language model trained on 100M lines of English monolingual data from WMT16-18 also contributed to rescoring the translations for this language direction.

6 Results and Evaluation

A critical component of our system is continued training (CT). To demonstrate the effectiveness of this method, we continue training using newstest2014-18, excluding newstest2017, using the learning rates mentioned in the previous section. The scores presented in Table 3 are reported on newstest2017.

Ensembling multiple models is a common way

System	Before CT	CT	CT-Ensemble
De-En	37.3	38.3	39.0 (x3)
En-De	30.8	32.3	32.6 (x2)

Table 3: Effect of continued training and ensembling, reported on newstest2017.

to improve performance of a NMT system. In Table 3, we observe a +0.74 improvement when ensembling 3 models (De-En) and +0.38 when ensembling 2 models (En-De).

M1	M2	M3	M1+M3	Ensemble (all)
30.8	29.7	30.8	32.6	32.3

Table 4: Results of ensembling En-De models, reported on newstest2017. Ensembling with the lower-performing model #2 (M2) degrades performance versus ensembling only models #1 and #3 (M1 and M3).

Table 4 shows the effects of ensembling En-De models with identical training setups, labeled M1, M2, and M3. M2 converged earlier than expected, and we observe that ensembling with this lower-performing model causes lower BLEU score than just ensembling the better performing models. As such, we exclude M2 from the final submission.

System	Our Submission	Highest Score
De-En	38.1	42.8 (MSRA)
En-De	42.5	44.9 (MSRA)

Table 5: BLEU-cased score on newstest2019.

For submission, we perform continued training using newstest2014-18 and ensemble multiple models with the same vocabulary for translation. We then employ reranking models on the 12-best lists produced from the ensembles.

6.1 Post-Submission Work

We built additional En-De and De-En translation models using the same training regimen described in this work. This allowed use to ensemble more models to boost performance. Results are seen in Table 6. Each post-submission ensemble was comprised of four models.

System	Submission Score	Final Score
De-En	38.1	**38.4**
En-De	42.5	**42.8**

Table 6: BLEU-cased score on newstest2019.

7 Conclusion

We began by replicating various top-scoring submissions from WMT 2018 (Bojar et al., 2018): Microsoft (Junczys-Dowmunt, 2018) and FAIR (Edunov et al., 2018). We were unable to match all the numbers from latter, perhaps due to our limited compute and differing hyperparameters.

Our system is built on various components from these submissions and JHU's 2018 submission (Koehn et al., 2018). We use clean bitext to train a backtranslation model (Transformer-big) and translate monolingual data using sampling inference. We filter the backtranslations, ParaCrawl, and Common Crawl, according to the domain and adequacy scores described in Junczys-Dowmunt (2018). We concatenate the filtered data with upsampled clean bitext to train Transformer-base translation models, and perform continued training over previous years' test sets.

An ensemble of such models are used to decode the test set, and translations are reranked using reranking models (Transformer-base) that are trained on a concatenation of upsampled bitext and filtered beam search backtranslated data. The reranking models also undergo equivalent continued training. On the De-En side, we also use a language model trained on 100 million monolingual English sentences to this effect. At the time of submission, we achieve a BLEU score of 38.1 for De-En and 42.5 for En-De. Our post-submission system consisting of 4-model ensembles scores 38.4 for De-En and 42.8 for En-De.

It is likely that effective training of Transformer-big models would have further boosted scores for our system, had we been able to do so on our single-GPU setup in time for this year's shared task.

Acknowledgements

The authors would like to thank Huda Khayrallah and Dan Povey for assistance with this project. We also appreciate the advice of Marcin Junczys-Dowmunt on filtering and hyperparameter optimization, Brian Thompson for guidance regarding continued training, and Mitchell Gordon for help with post-processing. We also thank our anonymous reviewers for their helpful comments.

References

Ondřej Bojar, Christian Federmann, Mark Fishel, Yvette Graham, Barry Haddow, Philipp Koehn, and Christof Monz. 2018. Findings of the 2018 conference on machine translation (WMT18). In *Proceedings of the Third Conference on Machine Translation: Shared Task Papers*, pages 272–303, Belgium, Brussels. Association for Computational Linguistics.

Ondřej Bojar, Christian Federmann, Mark Fishel, Yvette Graham, Barry Haddow, Matthias Huck, Philipp Koehn, Christof Monz, Mathias Müller, and Matt Post. 2019. Findings of the 2019 conference on machine translation (wmt19). In *Proceedings of the Fourth Conference on Machine Translation, Volume 2: Shared Task Papers*, Florence, Italy. Association for Computational Linguistics.

Sergey Edunov, Myle Ott, Michael Auli, and David Grangier. 2018. Understanding Back-Translation at Scale. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 489–500, Brussels, Belgium. Association for Computational Linguistics.

Vu Cong Duy Hoang, Philipp Koehn, Gholamreza Haffari, and Trevor Cohn. 2018. Iterative back-translation for neural machine translation. In *Proceedings of the 2nd Workshop on Neural Machine Translation and Generation*, pages 18–24, Melbourne, Australia. Association for Computational Linguistics.

Marcin Junczys-Dowmunt. 2018. Microsoft's Submission to the WMT2018 News Translation Task: How I Learned to Stop Worrying and Love the Data. In *Proceedings of the Third Conference on Machine Translation: Shared Task Papers*, pages 425–430, Belgium, Brussels. Association for Computational Linguistics.

Marcin Junczys-Dowmunt, Roman Grundkiewicz, Tomasz Dwojak, Hieu Hoang, Kenneth Heafield, Tom Neckermann, Frank Seide, Ulrich Germann, Alham Fikri Aji, Nikolay Bogoychev, André F. T. Martins, and Alexandra Birch. 2018. Marian: Fast Neural Machine Translation in C++. In *Proceedings of ACL 2018, System Demonstrations*, pages 116–121, Melbourne, Australia. Association for Computational Linguistics.

Philipp Koehn, Kevin Duh, and Brian Thompson. 2018. The JHU Machine Translation Systems for WMT 2018. In *Proceedings of the Third Conference on Machine Translation: Shared Task Papers*, pages 438–444.

Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, Chris Dyer, Ondřej Bojar, Alexandra Constantin, and Evan Herbst. 2007. Moses: Open Source Toolkit for Statistical Machine Translation. In *Proceedings of the 45th Annual Meeting of the ACL on Interactive Poster and Demonstration Sessions*, ACL '07.

Minh-Thang Luong and Christopher D. Manning. 2015. Neural Machine Translation Systems for Spoken Language Domains. In *International Workshop on Spoken Language Translation*.

Robert C. Moore and William Lewis. 2010. Intelligent Selection of Language Model Training Data. In *Proceedings of the ACL 2010 Conference Short Papers*, ACLShort '10, pages 220–224, Stroudsburg, PA, USA. Association for Computational Linguistics.

Myle Ott, Sergey Edunov, Alexei Baevski, Angela Fan, Sam Gross, Nathan Ng, David Grangier, and Michael Auli. 2019. fairseq: A Fast, Extensible Toolkit for Sequence Modeling. In *Proceedings of NAACL-HLT 2019: Demonstrations*.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. BLEU: A Method for Automatic Evaluation of Machine Translation. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, ACL '02, pages 311–318.

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. *arXiv preprint arXiv:1708.00726*.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. 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, Berlin, Germany. Association for Computational Linguistics.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is All you Need. In *Advances in Neural Information Processing Systems*, pages 5998–6008.