Bi-Directional Neural Machine Translation with Synthetic Parallel Data

Xing Niu
University of Maryland
xingniu@cs.umd.edu

Michael Denkowski
Amazon.com, Inc.
mdenkows@amazon.com

Marine Carpuat
University of Maryland
marine@cs.umd.edu

Abstract

Despite impressive progress in high-resource settings, Neural Machine Translation (NMT) still struggles in low-resource and out-of-domain scenarios, often failing to match the quality of phrase-based translation. We propose a novel technique that combines back-translation and multilingual NMT to improve performance in these difficult cases. Our technique trains a single model for both directions of a language pair, allowing us to back-translate source or target monolingual data without requiring an auxiliary model. We then continue training on the augmented parallel data, enabling a cycle of improvement for a single model that can incorporate any source, target, or parallel data to improve both translation directions. As a byproduct, these models can reduce training and deployment costs significantly compared to uni-directional models. Extensive experiments show that our technique outperforms standard back-translation in low-resource scenarios, improves quality on cross-domain tasks, and effectively reduces costs across the board.

1 Introduction

Neural Machine Translation (NMT) has been rapidly adopted in industry as it consistently outperforms previous methods across domains and language pairs (Bojar et al., 2017; Cettolo et al., 2017). However, NMT systems still struggle compared to Phrase-based Statistical Machine Translation (SMT) in low-resource or out-of-domain scenarios (Koehn and Knowles, 2017). This performance gap is a significant roadblock to full adoption of NMT.

In many low-resource scenarios, parallel data is prohibitively expensive or otherwise impractical to collect, whereas monolingual data may be more abundant. SMT systems have the advantage of a dedicated language model that can incorporate all available target-side monolingual data to significantly improve translation quality (Koehn et al., 2003; Koehn and Schroeder, 2007). By contrast, NMT systems consist of one large neural network that performs full sequence-to-sequence translation (Sutskever et al., 2014; Cho et al., 2014). Trained end-to-end on parallel data, these models lack a direct avenue for incorporating monolingual data. Sennrich et al. (2016a) overcome this challenge by back-translating target monolingual data to produce *synthetic* parallel data that can be added to the training pool. While effective, back-translation introduces the significant cost of first building a reverse system.

Another technique for overcoming a lack of data is multitask learning, in which domain knowledge can be transferred between related tasks (Caruana, 1997). Johnson et al. (2017) apply the idea to multilingual NMT by concatenating parallel data of various language pairs and marking the source with the desired output language. The authors report promising results for translation between languages that have zero parallel data. This approach also dramatically reduces the complexity of deployment by packing multiple language pairs into a single model.

We propose a novel combination of back-translation and multilingual NMT that trains both directions of a language pair jointly in a single model. Specifically, we initialize a bi-directional model on parallel data and then use it to translate select source and target monolingual data. Training is then continued on the augmented parallel data, leading to a cycle of improvement. This approach has several advantages:

- A single NMT model with standard architecture that performs all forward and backward translation during training.
- Training costs reduced significantly compared to uni-directional systems.
- Improvements in translating quality for low-resource languages, even over uni-directional systems with back-translation.
- Effectiveness in domain adaptation.

Via comprehensive experiments, we also contribute to best practices in selecting most suitable combinations of synthetic parallel data and choosing appropriate amount of monolingual data.

2 Approach

In this section, we introduce an efficient method for improving bi-directional neural machine translation with synthetic parallel data. We also present a strategy for selecting suitable monolingual data for back-translation.

2.1 Bi-Directional NMT with Synthetic Parallel Data

We use the techniques described by Johnson et al. (2017) to build a multilingual model that combines forward and backward directions of a single language pair. To begin, we construct training data by swapping the source and target sentences of a parallel corpus and appending the swapped version to the original. We then add an artificial token to the beginning of each source sentence to mark the desired target language, such as <2en> for English. A standard NMT system can then be trained on the augmented dataset, which is naturally balanced between language directions.[1] A shared Byte-Pair Encoding (BPE) model is built on source and target data, alleviating the issue of unknown words and reducing the vocabulary to a smaller set of items shared across languages (Sennrich et al., 2016b; Johnson et al., 2017). We further reduce model complexity by tying source and target word embeddings. The full training process significantly saves the total computing resources compared to training an individual model for each language direction.

Generating synthetic parallel data is straightforward with a bi-directional model: sentences from both source and target monolingual data can be translated to produce synthetic sentence pairs. Synthetic parallel data of the form `synthetic` $\rightarrow$ `monolingual` can then be used in the forward direction, the backward direction, or both. Crucially, this approach leverages both source and target monolingual data while always placing the real data on the target side, eliminating the need for work-arounds such as freezing certain model parameters to avoid degradation from training on MT output (Zhang and Zong, 2016).

2.2 Monolingual Data Selection

Given the goal of improving a base bi-directional model, selecting ideal monolingual data for back-translation presents a significant challenge. Data too close to the original training data may not provide sufficient new information for the model. Conversely, data too far from the original data may be translated too poorly by the base model to be useful. We manage these risks by leveraging a standard pseudo in-domain data selection technique, cross-entropy difference (Moore and Lewis, 2010; Axelrod et al., 2011), to rank sentences from a general domain. Smaller cross-entropy difference indicates a sentence that is simultaneously more similar to the in-domain corpus (e.g. real parallel data) and less similar to the average of the general-domain monolingual corpus. This allows us to begin with "safe" monolingual data and incrementally expand to higher risk but potentially more informative data.

3 Experiments

In this section, we describe data, settings, and experimental methodology. We then present the results of comprehensive experiments designed to answer the following questions: (1) How can synthetic data be most effectively used to improve translation quality? (2) Does the reduction in training time for bi-directional NMT come at the cost of lower translation quality? (3) Can we further improve training speed and translation quality training with incremental training and re-decoding? (4) How can we effectively choose monolingual training data? (5) How well does bi-directional NMT perform on domain adaptation?

3.1 Data

Diverse Language Pairs: We evaluate our approach on both high and low-resource data sets:

[1] Johnson et al. (2017) report the need to oversample when data is significantly unbalanced between language pairs.

Type	Dataset	# Sentences
High-resource: German↔English		
Training	Common Crawl + Europarl v7 + News Comm. v12	4,356,324
Dev	Newstest 2015+2016	5,168
Test	Newstest 2017	3,004
Mono-DE	News Crawl 2016	26,982,051
Mono-EN	News Crawl 2016	18,238,848
Low-resource: Tagalog↔English		
Training	News/Blog	50,705
Dev/Test	News/Blog	491/508
Dev/Test*	Bible	500/500
Sample*	Bible	61,195
Mono-TL	Common Crawl	26,788,048
Mono-EN	ICWSM 2009 blog	48,219,743
Low-resource: Swahili↔English		
Training	News/Blog	23,900
Dev/Test	News/Blog	491/509
Dev/Test*	Bible-NT	500/500
Sample*	Bible-NT	14,699
Mono-SW	Common Crawl	12,158,524
Mono-EN	ICWSM 2009 blog	48,219,743

Table 1: Data sizes of training, development, test, sample and monolingual sets. Sample data serves as the in-domain seed for data selection.

German↔English (DE↔EN), Tagalog↔English TL↔EN, and Swahili↔English (SW↔EN). Parallel and monolingual DE↔EN data are provided by the WMT17 news translation task (Bojar et al., 2017). Parallel data for TL↔EN and SW↔EN contains a mixture of domains such as news and weblogs, and is provided as part of the IARPA MATERIAL program.[2] We split the original corpora into training, dev, and test sets, therefore they share a homogeneous n-gram distribution. For these low-resource pairs, TL and SW monolingual data are provided by the Common Crawl (Buck et al., 2014) while EN monolingual data is provided by the ICWSM 2009 Spinn3r blog dataset (tier-1) (Burton et al., 2009).

Diverse Domain Settings: For WMT17 DE↔EN, we choose news articles from 2016 (the closest year to the test set) as in-domain data for back-translation. For TL↔EN and SW↔EN, we identify in-domain and out-of-domain monolingual data and apply data selection to choose pseudo in-domain data (see Section 2.2). We use the training data as in-domain and either Common Crawl or ICWSM as out-of-domain. We also include a low-resource, long-distance domain adaptation task for these languages: training on News/Blog data and testing on Bible data. We split a parallel Bible corpus (Christodoulopoulos and Steedman, 2015) into sample, dev, and test sets, using the sample data as the in-domain seed for data selection.

Preprocessing: Following Hieber et al. (2017), we apply four pre-processing steps to parallel data: normalization, tokenization, sentence-filtering (length 80 cutoff), and joint source-target BPE with 50,000 operations (Sennrich et al., 2016b). Low-resource language pairs are also true-cased to reduce sparsity. BPE and true-casing models are rebuilt whenever the training data changes. Monolingual data for low-resource settings is filtered by retaining sentences longer than nine tokens. Itemized data statistics after pre-processing can be found in Table 1.

3.2 NMT Configuration

We use the attentional RNN encoder-decoder architecture implemented in the Sockeye toolkit (Hieber et al., 2017). Our translation model uses a bi-directional encoder with a single LSTM layer of size 512, multilayer perceptron attention with a layer size of 512, and word representations of size 512 (Bahdanau et al., 2015). We apply layer normalization (Ba et al., 2016) and tie source and target embedding parameters. We train using the Adam optimizer with a batch size of 64 sentences and checkpoint the model every 1000 updates (10,000 for DE↔EN) (Kingma and Ba, 2015). Training stops after 8 checkpoints without improvement of perplexity on the development set. We decode with a beam size of 5.

For TL↔EN and SW↔EN, we add dropout to embeddings and RNNs of the encoder and decoder with probability 0.2. We also tie the output layer's weight matrix with the source and target embeddings to reduce model size (Press and Wolf, 2017). The effectiveness of tying input/output target embeddings has been verified on several low-resource language pairs (Nguyen and Chiang, 2018).

For TL↔EN and SW↔EN, we train four randomly seeded models for each experiment and combine them in a linear ensemble for decod-

[2] https://www.iarpa.gov/index.php/research-programs/material

ID	Training Data	TL→EN	EN→TL	SW→EN	EN→SW	DE→EN	EN→DE
U-1	L1→L2	31.99	31.28	32.60	39.98	29.51	23.01
U-2	L1→L2 + L1⋆→L2	**24.21**	**29.68**	**25.84**	**38.29**	**33.20**	**25.41**
U-3	L1→L2 + L1→L2⋆	22.13	27.14	24.89	36.53	30.89	23.72
U-4	L1→L2 + L1⋆→L2 + L1→L2⋆	23.38	29.31	25.33	37.46	33.01	25.05
	L1=EN	L2=TL		L2=SW		L2=DE	
B-1	L1↔L2	32.72	31.66	33.59	39.12	28.84	22.45
B-2	L1↔L2 + L1⋆↔L2	32.90	**32.33**	33.70	**39.68**	29.17	**24.45**
B-3	L1↔L2 + L2⋆↔L1	32.71	31.10	33.70	39.17	**31.71**	21.71
B-4	L1↔L2 + L1⋆↔L2 + L2⋆↔L1	**33.25**	**32.46**	**34.23**	38.97	30.43	22.54
B-5	L1↔L2 + L1⋆→L2 + L2⋆→L1	**33.41**	**33.21**	**34.11**	**40.24**	**31.83**	**24.61**
B-5*	L1↔L2 + L1⋆→L2 + L2⋆→L1	33.79	32.97	34.15	40.61	31.94	24.45
B-6*	L1↔L2 + L1⋆→L2 + L2⋆→L1	**34.50**	**33.73**	**34.88**	**41.53**	**32.49**	**25.20**

Table 2: BLEU scores for uni-directional models (U-*) and bi-directional NMT models (B-*) trained on different combinations of real and synthetic parallel data. Models in B-5* are fine-tuned from base models in B-1. Best models in B-6* are fine-tuned from precedent models in B-5* and underscored synthetic data is re-decoded using precedent models. Scores with largest improvement within each zone are highlighted.

ing. For DE↔EN experiments, we train a single model and average the parameters of the best four checkpoints for decoding (Junczys-Dowmunt et al., 2016). We report case-insensitive BLEU with standard WMT tokenization.[3]

3.3 Uni-Directional NMT

We first evaluate the impact of synthetic parallel data on standard uni-directional NMT. Baseline systems trained on real parallel data are shown in row U-1 of Table 2.[4] In all tables, we use L1→L2 to indicate real parallel data where the source language is L1 and the target language is L2. Synthetic data is annotated by asterisks, such as L1⋆→L2 indicating that L1⋆ is the synthetic back-translation of real monolingual data L2.

We always select monolingual data as an integer multiple of the amount of real parallel data n, i.e. $|\text{L1}\rightarrow\text{L2}\star| = |\text{L1}\star\rightarrow\text{L2}| = kn$. For DE↔EN models, we simply choose the top-n sentences from shuffled News Crawl corpus. For all models of low-resource languages, we select the top-$3n$ sentences ranked by cross-entropy difference as described in Section 2.2. The choice of k is discussed in Section 3.4.2.

Shown in rows U-2 through U-4 of Table 2, we compare the results of incorporating different combinations of real and synthetic parallel data. Models trained on **only real data of target language** (i.e. in U-2) achieve better performance in BLEU than using other combinations. This is an expected result since translation quality is highly correlated with target language models. By contrast, standard back-translation is not effective for our low-resource scenarios. A significant drop (∼7 BLEU comparing U-1 and U-2 for TL/SW→EN) is observed when back-translating English. One possible reason is that the quality of the selected monolingual data, especially English, is not ideal. We will encounter this issue again when using bi-directional models with the same data in Section 3.4.

3.4 Bi-Directional NMT

We map the same synthetic data combinations to bi-directional NMT, comparing against uni-directional models with respect to both translation quality and training time. Training bi-directional models requires doubling the training data by adding a second copy of the parallel corpus where the source and target are swapped. We use the notation L1↔L2 to represent the concatenation of L1→L2 and its swapped copy L2→L1 in Table 2.

Compared to independent models (i.e. U-1), the bi-directional DE↔EN model in B-1 is slightly worse (by ∼0.6 BLEU). These losses match observations by Johnson et al. (2017) on many-to-many multilingual NMT models. By contrast, bi-directional low-resource models slightly outper-

[3] We use the script https://github.com/EdinburghNLP/nematus/blob/master/data/multi-bleu-detok.perl

[4] Baseline BLEU scores are higher than expected on low-resource language pairs. We hypothesize that the data is homogeneous and easier to translate.

Model		TL→EN	EN→TL	SW→EN	EN→SW	DE→EN	EN→DE
Uni-directional	Baseline	76	78	63	66	41	48
	Synthetic	177	176	137	104	88	75
	TOTAL		507		371		252
Bi-directional	Baseline		125		93		61
	Synthetic		285		218		113
	TOTAL	↓ 19%	410	↓ 14%	311	↓ 31%	174
(fine-tuning)	Synthetic	↓ 23%	219	↓ 44%	122	↓ 24%	86

Table 3: Number of checkpoints (= |updates|/1000 for TL/SW↔EN or |updates|/10,000 for DE↔EN) used by various NMT models. Bi-directional models reduce the training time by 15-30% (comparing 'TOTAL' rows). Fine-tuning bi-directional baseline models on synthetic parallel data reduces the training time by 20-40% (comparing 'Synthetic' rows).

form independent models. We hypothesize that in low-resource scenarios the neural model's capacity is far from exhausted due to the redundancy in neural network parameters (Denil et al., 2013), and the benefit of training on twice as much data surpasses the detriment of confusing the model by mixing two languages.

We generate synthetic parallel data from the same monolingual data as in the uni-directional experiments. If we build training data symmetrically (i.e. B-2,3,4), back-translated sentences are distributed equally on the source and target sides, forcing the model to train on some amount of synthetic target data (MT output). For DE↔EN models, the best BLEU scores are achieved when synthetic training data is only present on the source side while for low-resource models, the results are mixed. We see a particularly counter-intuitive result when using monolingual English data — no significant improvement (see B-3 for TL/SW→EN). As bi-directional models are able to leverage monolingual data of both languages, better results are achieved when combining all synthetic parallel data (see B-4 for TL/SW→EN). By further excluding potentially harmful target-side synthetic data (i.e. B-4 → B-5), the most unified and slim models achieve best overall performance.

While the best bi-directional NMT models thus far (B-5) outperform the best uni-directional models (U-1,2) for low-resource language pairs, they struggle to match performance in the high-resource DE↔EN scenario.

In terms of efficiency, bi-directional models consistently reduce the training time by 15-30% as shown in Table 3. Note that checkpoints are summed over all independent runs when ensemble decoding is used.

3.4.1 Fine-Tuning and Re-Decoding

Training new NMT models from scratch after generating synthetic data is incredibly expensive, working against our goal of reducing the overall cost of deploying strong translation systems. Following the practice of mixed fine-tuning proposed by Chu et al. (2017), we continue training baseline models on augmented data as shown in B-5* of Table 2. These models achieve comparable translation quality to those trained from scratch (B-5) at a significantly reduced cost, up to 20-40% computing time in the experiments illustrated in Table 3.

We also explore re-decoding the same monolingual data using improved models (Sennrich et al., 2016a). Underscored synthetic data in B-6* is re-decoded by models in B-5*, leading to the best results for all low-resource scenarios and competitive results for our high-resource scenario.

3.4.2 Size of Selected Monolingual Data

In our experiments, the optimal amount of monolingual data for constructing synthetic parallel data is task-dependent. Factors such as size and linguistic distribution of data and overlap between real parallel data, monolingual data, and test data can influence the effectiveness curve of synthetic data. We illustrate the impact of varying the size of selected monolingual data in our low-resource scenario. Shown in Figure 1, all language pairs have an increasing momentum and tend to converge with more synthetic parallel data. The optimal point is a hyper-parameter that can be empirically determined.

3.4.3 Domain Adaptation

We evaluate the performance of using the same bi-directional NMT framework on a long-distance

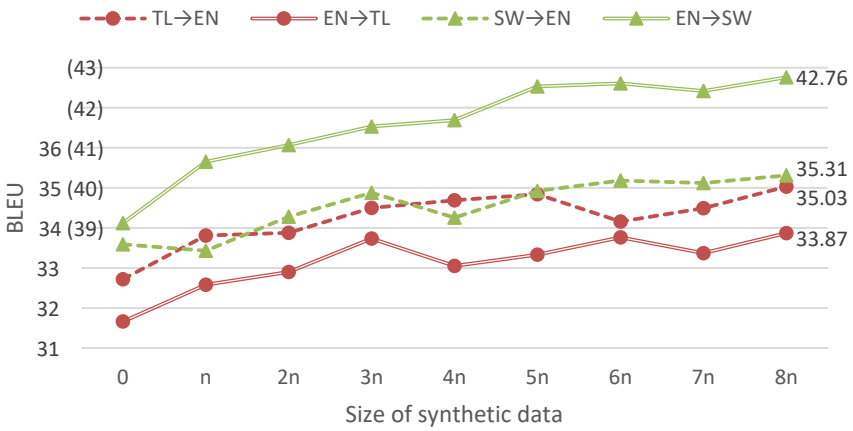

Figure 1: BLEU scores for four translation directions vs. the size of selected monolingual data. n in x-axis equals to the size of real parallel data. EN→SW models use BLEU in parentheses in y-axis. All language pairs have an increasing momentum and tend to converge with more synthetic parallel data.

		L2=TL		L2=SW	
ID	Training Data (L1=EN)	TL→EN	EN→TL	SW→EN	EN→SW
A-1	L1↔L2	11.03	10.17	6.56	3.80
A-2	L1↔L2 + L1⋆→L2 + L2⋆→L1	16.49	22.33	8.70	7.47
A-3	L1↔L2 + L1⋆→L2 + L2⋆→L1	**18.91**	**23.41**	**11.01**	**8.06**

Table 4: BLEU scores for bi-directional NMT models on Bible data. Models in A-2 are fine-tuned from baseline models in A-1. Highlighted best models in A-3 are fine-tuned from precedent models in A-2 and underscored synthetic data is re-decoded using precedent models. Baseline models are significantly improved in terms of BLEU.

domain adaptation task: News/Blog to Bible. This task is particularly challenging because out-of-vocabulary rates of Bible test sets are as high as 30-45% when training on News/Blog. Significant linguistic differences also exist between modern and Biblical language use. The impact of this domain mismatch is demonstrated by the incredibly low BLEU scores of baseline News/Blog systems (Table 4, A-1). After fine-tuning baseline models on augmented parallel data (A-2) and re-decoding (A-3),[5] we see BLEU scores increase by 70-130%. Despite being based on extremely weak baseline performance, they still show the promise of our approach for domain adaptation.

4 Related Work

Leveraging monolingual data in NMT is challenging. For example, integrating language models in the decoder (Gülçehre et al., 2015) or initializing the encoder and decoder with pre-trained language models (Ramachandran et al., 2017) would require significant changes to system architecture.

In this work, we build on the elegant and effective approach of turning incomplete (monolingual) data into complete (parallel) data by back-translation. Sennrich et al. (2016a) used an auxiliary reverse-directional NMT system to generate synthetic source data from real monolingual target data, with promising results (+3 BLEU on strong baselines). Symmetrically, Zhang and Zong (2016) used an auxiliary same-directional translation system to generate synthetic target data from the real source language. However, parameters of the decoder have to be frozen while training on synthetic data, otherwise the decoder would fit to noisy MT output. By contrast, our approach effectively leverages synthetic data from both translation directions, with consistent gains in translation quality. A similar idea is used by Zhang et al. (2018) with a focus on re-decoding iteratively. However, their NMT models of both directions are still trained independently.

Another technique for using monolingual data in NMT is round-trip machine translation. Sup-

[5]The concatenation of development sets from both News/Blog and Bible serves for validation.

pose sentence f from a monolingual dataset is translated forward to e and then translated back to f', then f' and f should be identical (Brislin, 1970). Cheng et al. (2016) optimize $\arg\max_\theta P(f'|f;\theta)$ as an autoencoder; Wang et al. (2018) minimize the difference between $P(f)$ and $P(f'|\theta)$ based on the law of total probability, while He et al. (2016) set the quality of both e and f' as rewards for reinforcement learning. They all achieve promising improvement but rely on non-standard training frameworks.

Multitask learning has been used in past work to combine models trained on different parallel corpora by sharing certain components. These components, such as the attention mechanism (Firat et al., 2016), benefit from being trained on an effectively larger dataset. In addition, the more parameters are shared, the faster a joint model can be trained — this is particularity beneficial in industry settings. Baidu built one-to-many translation systems by sharing both encoder and attention (Dong et al., 2015). Google enabled a standard NMT framework to support many-to-many translation directions by simply attaching a language specifier to each source sentence (Johnson et al., 2017). We adopted Google's approach to build bi-directional systems that successfully combine actual and synthetic parallel data.

5 Conclusion

We propose a novel technique for bi-directional neural machine translation. A single model with a standard NMT architecture performs both forward and backward translation, allowing it to back-translate and incorporate any source or target monolingual data. By continuing training on augmented parallel data, bi-directional NMT models consistently achieve improved translation quality, particularly in low-resource scenarios and cross-domain tasks. These models also reduce training and deployment costs significantly compared to standard uni-directional models.

Acknowledgments

Part of this research was conducted while the first author was an intern at Amazon. At Maryland, this research is based upon work supported in part by the Clare Boothe Luce Foundation, and by the Office of the Director of National Intelligence (ODNI), Intelligence Advanced Research Projects Activity (IARPA), via contract #FA8650-17-C-9117. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies, either expressed or implied, of ODNI, IARPA, or the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for governmental purposes notwithstanding any copyright annotation therein.

References

Amittai Axelrod, Xiaodong He, and Jianfeng Gao. 2011. Domain adaptation via pseudo in-domain data selection. In *EMNLP*, pages 355–362. ACL.

Lei Jimmy Ba, Ryan Kiros, and Geoffrey E. Hinton. 2016. Layer normalization. *CoRR*, abs/1607.06450.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2015. Neural machine translation by jointly learning to align and translate. In *ICLR*.

Ondrej Bojar, Rajen Chatterjee, Christian Federmann, Yvette Graham, Barry Haddow, Shujian Huang, Matthias Huck, Philipp Koehn, Qun Liu, Varvara Logacheva, Christof Monz, Matteo Negri, Matt Post, Raphael Rubino, Lucia Specia, and Marco Turchi. 2017. Findings of the 2017 conference on machine translation (WMT17). In *WMT*, pages 169–214. Association for Computational Linguistics.

Richard W Brislin. 1970. Back-translation for cross-cultural research. *Journal of cross-cultural psychology*, 1(3):185–216.

Christian Buck, Kenneth Heafield, and Bas van Ooyen. 2014. N-gram counts and language models from the common crawl. In *LREC*, pages 3579–3584. European Language Resources Association (ELRA).

Kevin Burton, Akshay Java, and Ian Soboroff. 2009. The ICWSM 2009 Spinn3r dataset. In *Proceedings of the Third Annual Conference on Weblogs and Social Media (ICWSM 2009), San Jose, CA*.

Rich Caruana. 1997. Multitask learning. *Machine Learning*, 28(1):41–75.

Mauro Cettolo, Marcello Federico, Luisa Bentivogli, Jan Niehues, Sebastian Stüker, Katsuitho Sudoh, Koichiro Yoshino, and Christian Federmann. 2017. Overview of the IWSLT 2017 evaluation campaign. In *International Workshop on Spoken Language Translation*, pages 2–14.

Yong Cheng, Wei Xu, Zhongjun He, Wei He, Hua Wu, Maosong Sun, and Yang Liu. 2016. Semi-supervised learning for neural machine translation. In *ACL (1)*. The Association for Computer Linguistics.

Kyunghyun Cho, Bart van Merrienboer, Çaglar Gülçehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning phrase representations using RNN encoder-decoder for statistical machine translation. In *EMNLP*, pages 1724–1734. ACL.

Christos Christodoulopoulos and Mark Steedman. 2015. A massively parallel corpus: the bible in 100 languages. *Language Resources and Evaluation*, 49(2):375–395.

Chenhui Chu, Raj Dabre, and Sadao Kurohashi. 2017. An empirical comparison of domain adaptation methods for neural machine translation. In *ACL (2)*, pages 385–391. Association for Computational Linguistics.

Misha Denil, Babak Shakibi, Laurent Dinh, Marc'Aurelio Ranzato, and Nando de Freitas. 2013. Predicting parameters in deep learning. In *NIPS*, pages 2148–2156.

Daxiang Dong, Hua Wu, Wei He, Dianhai Yu, and Haifeng Wang. 2015. Multi-task learning for multiple language translation. In *ACL (1)*, pages 1723–1732. The Association for Computer Linguistics.

Orhan Firat, Kyunghyun Cho, and Yoshua Bengio. 2016. Multi-way, multilingual neural machine translation with a shared attention mechanism. In *HLT-NAACL*, pages 866–875. The Association for Computational Linguistics.

Çaglar Gülçehre, Orhan Firat, Kelvin Xu, Kyunghyun Cho, Loïc Barrault, Huei-Chi Lin, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2015. On using monolingual corpora in neural machine translation. *CoRR*, abs/1503.03535.

Di He, Yingce Xia, Tao Qin, Liwei Wang, Nenghai Yu, Tie-Yan Liu, and Wei-Ying Ma. 2016. Dual learning for machine translation. In *NIPS*, pages 820–828.

Felix Hieber, Tobias Domhan, Michael Denkowski, David Vilar, Artem Sokolov, Ann Clifton, and Matt Post. 2017. Sockeye: A toolkit for neural machine translation. *CoRR*, abs/1712.05690.

Melvin Johnson, Mike Schuster, Quoc V. Le, Maxim Krikun, Yonghui Wu, Zhifeng Chen, Nikhil Thorat, Fernanda B. Viégas, Martin Wattenberg, Greg Corrado, Macduff Hughes, and Jeffrey Dean. 2017. Google's multilingual neural machine translation system: Enabling zero-shot translation. *TACL*, 5:339–351.

Marcin Junczys-Dowmunt, Tomasz Dwojak, and Rico Sennrich. 2016. The AMU-UEDIN submission to the WMT16 news translation task: Attention-based NMT models as feature functions in phrase-based SMT. In *WMT*, pages 319–325. The Association for Computer Linguistics.

Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *ICLR*.

Philipp Koehn and Rebecca Knowles. 2017. Six challenges for neural machine translation. In *NMT@ACL*, pages 28–39. Association for Computational Linguistics.

Philipp Koehn, Franz Josef Och, and Daniel Marcu. 2003. Statistical phrase-based translation. In *HLT-NAACL*. The Association for Computational Linguistics.

Philipp Koehn and Josh Schroeder. 2007. Experiments in domain adaptation for statistical machine translation. In *WMT@ACL*, pages 224–227. Association for Computational Linguistics.

Robert C. Moore and William D. Lewis. 2010. Intelligent selection of language model training data. In *ACL (Short Papers)*, pages 220–224. The Association for Computer Linguistics.

Toan Q. Nguyen and David Chiang. 2018. Improving lexical choice in neural machine translation. In *HLT-NAACL*. The Association for Computational Linguistics.

Ofir Press and Lior Wolf. 2017. Using the output embedding to improve language models. In *EACL (2)*, pages 157–163. Association for Computational Linguistics.

Prajit Ramachandran, Peter J. Liu, and Quoc V. Le. 2017. Unsupervised pretraining for sequence to sequence learning. In *EMNLP*, pages 383–391. Association for Computational Linguistics.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016a. Improving neural machine translation models with monolingual data. In *ACL (1)*. The Association for Computer Linguistics.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016b. Neural machine translation of rare words with subword units. In *ACL (1)*. The Association for Computer Linguistics.

Ilya Sutskever, Oriol Vinyals, and Quoc V. Le. 2014. Sequence to sequence learning with neural networks. In *NIPS*, pages 3104–3112.

Yijun Wang, Yingce Xia, Li Zhao, Jiang Bian, Tao Qin, Guiquan Liu, and Tie-Yan Liu. 2018. Dual transfer learning for neural machine translation with marginal distribution regularization. In *AAAI*. AAAI Press.

Jiajun Zhang and Chengqing Zong. 2016. Exploiting source-side monolingual data in neural machine translation. In *EMNLP*, pages 1535–1545. The Association for Computational Linguistics.

Zhirui Zhang, Shujie Liu, Mu Li, Ming Zhou, and Enhong Chen. 2018. Joint training for neural machine translation models with monolingual data. In *AAAI*. AAAI Press.