Xiaomi's Submissions for IWSLT 2020 Open Domain Translation Task

Yuhui Sun, Mengxue Guo, Xiang Li, Jianwei Cui, Bin Wang
AI Lab, Xiaomi Group
{sunyuhui1,guomengxue1,lixiang21,cuijianwei,wangbin11}@xiaomi.com

Abstract

This paper describes the Xiaomi's submissions to the IWSLT20 shared open domain translation task for Chinese↔Japanese language pair. We explore different model ensembling strategies based on recent Transformer variants. We also further strengthen our systems via some first-line techniques, such as data filtering, data selection, tagged back translation, domain adaptation, knowledge distillation, and re-ranking. Our resulting Chinese→Japanese primary system ranked second in terms of character-level BLEU score among all submissions. Our resulting Japanese→Chinese primary system also achieved a competitive performance.

1 Introduction

In this paper, we describe the Xiaomi's neural machine translation (NMT) systems evaluated at IWSLT 2020 (Ansari et al., 2020) shared open domain translation task in two directions, Chinese→Japanese (Zh→Ja) and Japanese→Chinese (Ja→Zh).

The accuracy of NMT systems relies on the quality of training data, we first consider careful preprocessing and discard the corrupted data from the existing bilingual sentences according to rule-based filtering and model-based scoring.

In the aspect of NMT architecture, we exploit some recent Transformer variants, including different Transformer models with deeper layers or wider inner dimension of feed-forward layers than the standard Transformer-Big model, Transformer with a dynamic linear combination of layers (DLCL) (Wang et al., 2019) and neural architecture search (NAS) based Transformer-Evolved (So et al., 2019), to increase the diversity of the system. We further strengthen our systems by diversifying the training data via some effective methods, including back-translation (BT) (Sennrich et al., 2016b), knowledge distillation (KD) (Hinton et al., 2015) and right-to-left (R2L) NMT model. Finally, we also explore re-rank the n-best translation candidates generated by models ensembling with some effective features, including target-to-source (T2S) NMT model, left-to-right (L2R) NMT model, R2L NMT model (Liu et al., 2016), bilingual sentence BERT and language model (LM).

Through experiments, we evaluate how each system feature affects the accuracy of NMT. Our resulting Chinese→Japanese primary system ranked second in terms of character-level BLEU score among all submissions. Our resulting Japanese→Chinese primary system also achieved a competitive performance.

2 Data

2.1 Pre-processing

Our pre-processing pipeline begins by removing non-printable ASCII characters, lowercasing text, normalizing additional white-space, and control character and replacing any escaped characters with the corresponding symbol by our in-house script. All the data is further normalized so all full-width Roman characters and digits are normalized to half-width. All the traditional characters of Chinese data are converted to simplified characters using OpenCC[1]. For all corpora, Chinese sentences are segmented by our in-house Chinese word segmenter, and Japanese sentences are first segmented by the morphological analyzer Mecab (Kudo, 2006) and then tokenized only for the non-Japanese part by the Moses script[2].

[1] https://github.com/BYVoid/OpenCC
[2] https://github.com/moses-smt/mosesdecoder/blob/master/scripts/tokenizer/tokenizer.perl

Proceedings of the 17th International Conference on Spoken Language Translation (IWSLT), pages 149–157
July 9-10, 2020.
https://doi.org/10.18653/v1/P17

2.2 Parallel Data Filtering

Though the NMT performance is highly correlated to the huge amounts of training data, a robust body of studies (Carpuat et al., 2017; Khayrallah and Koehn, 2018; Wang et al., 2018; Koehn et al., 2018) has shown the bad impact of noisy data on general NMT translation accuracy. In addition to a small amount of Japanese-Chinese parallel data[3] from various public sources, the organizers also provide a large-scale but noisy parallel data[4] extracted from a non-parallel web-crawled data through some similarity measures for parallel data mining. We apply a two-stage process consisting of rule-based filtering and model-based scoring to further filter harmful sentence pairs that are bound to negatively affect the quality of NMT systems from the original parallel corpora as follows.

2.2.1 Rule-based Filtering

During the first stage, we remove some illegal parallel sentences by applying several rule-based heuristics. A sentence pair is deleted from the corpus if its source side or target side fails to obey any of the following wild rules reflecting what 'good data' should look like. Some of the heuristic filtering methods can deal with aspects that can not be captured with models.

- The token (i.e. character sequence between two spaces) length of every sentence is limited less than 50.
- Sentence pairs with a length ratio greater than 4 are removed.
- Chinese sentences with Chinese characters ratio less than 0.15 or any character of other than Chinese and English are removed. And Japanese sentences with Japanese characters ratio less than 0.25 or any character of other than Chinese, Japanese, and English are removed.
- Japanese sentences without any Hiragana or Katakana character are removed.
- Sentence pairs with mismatched numbers of length three or more digits or URLs are removed.
- Duplicated sentence pairs are discarded.

2.2.2 Model-based Scoring

In the second stage of our filtering pipeline, we utilize a variety of models to assign some scores to each sentence pair of the remaining rule-based filtered parallel corpus (RFPD). Afterward, we select better sentences according to these scores.

- Translation model: We construct parallel NMT systems based on the standard Transformer-big model in both directions using RFPD to obtain the target synthetic translation as the reference. BEER (Stanojević and Sima'an, 2014) is used as a sentence-level metric of sentence similarity. We prune the sentence pairs with the BEER score of lower than 0.2.
- SBERT model: Recently, contextualized word embeddings derived from large-scale pre-trained language models (Devlin et al., 2019; Liu et al., 2019; Yang et al., 2019) have achieved new state-of-the-arts in various monolingual NLP tasks. The success has also been extended to cross-lingual scenarios (Schwenk, 2018; Conneau and Lample, 2019; Mulcaire et al., 2019; Artetxe and Schwenk, 2019). Recently, Reimers and Gurevych (2019) proposed sentence BERT (SBERT) to derive semantically meaningful sentence embeddings. According to the training framework of SBERT, we use the multilingual pre-train BERT model[5] and finetune it on RFPD to yield useful Chinese and Japanese sentence embeddings in the same space. We reject sentence pairs with a cosine-similarity score below 0.2.
- Word alignment model: We perform a word alignment model on RFPD using *fast_align* (Dyer et al., 2013) to check whether the sentence pair has the same meaning. Sentence pairs with the alignment probability of being each other translation less than 0.1 are discarded.
- N-gram LM: It is beneficial to use fluent sentences for training NMT models. We train a 5-gram LM that is estimated with modified Kneser-Ney smoothing (Kneser and Ney,

[3] https://iwslt.oss-cn-beijing.aliyuncs.com/existing_parallel.tgz

[4] https://iwslt.oss-cn-beijing.aliyuncs.com/web_crawled_parallel_filtered_1.1.tgz

[5] https://storage.googleapis.com/bert_models/2018_11_23/multi_cased_L-12_H-768_A-12.zip

1995) using KenLM (Heafield, 2011) on each side of the parallel sentences to evaluate sentences' naturalness. We normalize the LM perplexity (PPL) scores of all the sentences to be between [0,1]. Sentences whose normalized PPL scores fall below the threshold (0.45 and 0.53 for Chinese and Japanese data, respectively) are removed.

It is worth noting that all the above thresholds are determined experimentally.

2.3 Post-processing

All the outputs are post-processed by merging subwords, removing the space between the non-ASCII characters, and rule-based de-truecasing. All half-width punctuation marks and digits are also converted back to their original full-width form in a specific language when translating to Chinese and Japanese.

3 Overview of System Features

3.1 Translation Models

NMT has gained rapid progress in recent years (Sutskever et al., 2014; Bahdanau et al., 2015; Vaswani et al., 2017). In addition to the standard Transformer-Big (Vaswani et al., 2017) model, we also apply recent Transformer variants for creating better model ensembles.

- Wider model: Dimension is an important factor to enhance the Transformer model capacity and performance. Based on the standard Transformer-Big model, we train a Transformer-Wide model with a inner dimension of position-wise feed-forward layers 8,192.

- Deeper model: Building deeper networks via stacking more encoder and decoder layers has been a trend in NMT (Bapna et al., 2018; Wu et al., 2019; Zhang et al., 2019). We also exploit three deeper Transformer models by simply increasing the layer size of Transformer-Big, including Transformer-Deep-12-12, Transformer-Deep-12-6, and Transformer-Deep-6-12 in which the first number represents the layer size of the encoder and the second number represents the layer size of the decoder. In addition to the standard Transformer in which the residual connection is applied between two adjacent layers, we also implement two DLCL (Wang et al., 2019)-based Transformer models which can memorize the outputs from all preceding layers, including Transformer-DLCL-Big based on the Transformer-Big model and Transformer-DLCL-Deep based on the Transformer-Deep-12-12 above.

- NAS-based model: Recently, NAS has begun to outperform human-designed models (Elsken et al., 2018). We use the computationally efficient Transformer-Evolved (So et al., 2019) model by NAS. The hyper-parameters can be seen in Tensor2Tensor implementation[6].

3.2 Data Diversification

We employ an effective data augmentation strategy to boost NMT accuracy by diversifying the training data. We first use the following backward and forward models to generate a diverse set of synthetic training data from both lingual sides of the original training data or external monolingual data. Then, we concatenate all the synthetic data with the original data to train the baseline models from scratch in L2R, R2L, and T2S ways, respectively. Finally, we conduct the aforementioned approach based on ensemble models again to achieve better baseline systems.

- T2S model: Back-translation has thus far been the most effective technique effective for NMT (Sennrich et al., 2016b). Instead of using the synthetic training data produced by translating monolingual data in the target language into the source language conventionally, we prepend a special tag to all the source sentences from the synthetic data to distinguish synthetic data from original data (Caswell et al., 2019).

- R2L model: Generally, most NMT systems produce translations in an L2R way, which suffers from the issue of exposure bias and consequent error propagation (Ranzato et al., 2016). It has been observed that the accuracy of the right part words in its translation results is usually worse than the left part words (Zhang et al., 2018; Zhou et al., 2019). We train all the baseline systems separately using L2R

[6] https://github.com/tensorflow/tensor2tensor/blob/master/tensor2tensor/models/evolved_transformer.py

and R2L decoding (Wang et al., 2017; Hassan et al., 2018).

- L2R model: Knowledge distillation has been widely applied to NMT (Kim and Rush, 2016; Freitag et al., 2017; Chen et al., 2017; Gu et al., 2018; Tan et al., 2019). Recent work (Furlanello et al., 2018) demonstrates that the student model can surpass the accuracy of the teacher model, even if the student model is identical to their teacher model. Following this work, the teacher and student models in our experiments keep the same architecture.

3.3 Model Ensembling

Ensemble decoding is an effective approach to boost the accuracy of NMT systems via averaging the word distributions output from multiple single models at each decoding step. We select the top 4 systems with the highest BLEU evaluated on the development dataset from all the available baseline systems of each direction for models ensembling.

3.4 Reranking

Reranking technique (Shen et al., 2004) has been applied in the recent years' WMT tasks (Sennrich et al., 2016a; Wang et al., 2017; Ng et al., 2019) and have provided significant improvements. We first use the S2T-L2R and S2T-R2L ensemble systems to generate more diverse translation hypotheses for a source sentence (Liu et al., 2016). Then we use ensemble models of S2T-L2R, S2T-R2L and T2S-L2R to calculate 3 different likelihood scores for each sentence pair. We obtain the perplexity score for the translation candidates with a neural LM based on the Transformer encoder. We also employ SBERT to calculate the similarity score for each sentence pair. Each model's score is treated as an individual feature. Considering the ranking problem as a classification problem, we employ the implementation of pairwise ranking in scikit-learn[7] RankSVM (Joachims, 2006) to learn the weights of all the features on the development data for reranking. We compute the relative distance between these two samples in the sentence-level BLEU metric by pairing up two translation candidates. In the training phase of the reranking model, we are only interested in whether the relative distance is positive or negative. For test data, we rescore the hypotheses in the list by the reranking model and select the hypothesis with the highest likelihood score as the final output.

4 Experiments and Results

In this section, we introduce the experimental and data setup used in our experiments and then evaluate each of the systems introduced in Section 3.

4.1 Experimental and Data Setup

Due to a large number of training parameters, our deeper Transformer models require larger GPU memory resources and more time to train. To avoid the out-of-memory issue when training models with adequate batch size, all models are optimized by the memory-efficient Adafactor (Shazeer and Stern, 2018) which has three times smaller models than Adam (Kingma and Ba, 2015). Furthermore, we also apply the mixed-precision training (Narang et al., 2018) without losing model accuracy to speed up the training significantly.

In the training stage, we batch sentence pairs by approximate length and limit the number of source and target tokens per batch to 2,048 for two deeper models and 4,096 for others per GPU. All models are trained on one machine with 8 NVIDIA V100 GPUs each of which has 16GB memory for a total of 200K steps. We optimize all models against BLEU using the development set provided by the organizer, stopping early if BLEU does not improve for 16 checkpoints of 2,000 updates each. We set dropout 0.1 for Chinese→Japanese and 0.2 is for Japanese→Chinese. We average the top 10 checkpoints evaluated against the development set as the final model for decoding. During decoding, the beam size is set to 4 for the single model and 10 for ensemble models. We report the 4-gram character BLEU (Papineni et al., 2002) evaluated by the provided automatic evaluation script[8].

The approach of two-stage parallel data filtering in Section 2.2 enables us to drastically reduce the training data from 19M to 12M. In order to enlarge the size of bilingual data, we also exploit to extract more high-quality sentence pairs from the provided pre-filtered parallel data[9]. We first pre-process the data and use the rules in Section 2.2 to remove

[7] https://scikit-learn.org/stable/modules/generated/sklearn.svm.LinearSVC.html

[8] https://github.com/didi/iwslt2020_open_domain_translation/blob/master/scripts/multi-bleu-detok.perl

[9] https://iwslt.oss-cn-beijing.aliyuncs.com/web_crawled_parallel_1.1.tgz

Corpus	#Sentences	Zh→Ja	Ja→Zh
Original	21M	32.24	26.03
Filtered	12M	36.94	31.27
+augment($\mathcal{D}_1$)	16M	37.08	31.44

Table 1: Results for L2R Transformer-Big based Chinese↔Japanese systems on the development dataset with different training data.

illegal data. We then rank the remaining data according to the sum of S2T and T2S BEER scores of each sentence pair and select 4M sentence pairs with the highest score into the filtered training data. Finally, we obtain the augmented training data with 16M sentence pairs to train all models.

We learn BPE segmentation models (Sennrich et al., 2016c) with 30K merge operations and filter out sentence pairs consisting of rare subword units with a frequency threshold of less than 6 to speed up the training, in which 38.5K and 40K subword tokens are adopted as Chinese and Japanese vocabularies separately for each experiment.

We submit two systems per direction in constrained and unconstrained training data settings. In a constrained condition, we only use the training data provided by the organizer. And for unconstrained submission, we choose the large-scale amounts of Commoncrawl Chinese[10] and Japanese[11] dataset as additional monolingual data for training LMs and executing BT to enhance our NMT systems. We process these monolingual data as follows: (1)pre-process according to the pipeline described in Section 2.1; (2)sentence segmentation; (3) only keep sentences with token length between 5 and 100; (4) draw a random sample with 160M sentences as the final clean monolingual data for each language.

4.2 Results of Data Filtering and Augmentation

We first evaluate the effect of data filtering on the performance of the NMT system. We train the Transformer-Big model on (i) the original training data only, (ii) filtered training data, (iii) concatenating selected 4M training data from the provided pre-filtered parallel data (+augment). Table 1 shows that data filtering gives a significant improvement for NMT accuracy, up to 4.70 BLEU score for Zh→Ja and 5.24 BLEU score for Ja→Zh, and adding more high-quality data can further boost the performance for Zh↔Ja. The results shed light on the importance of effective data filtering for training a strong NMT system, particularly for the training data with much noise mined from the web. Finally, $\mathcal{D}_1$ with 16M sentence pairs is chosen as the starting training data for the task.

4.3 Results of Baseline Models

For each translation task, we compare the performance of all the baseline systems trained on $\mathcal{D}_1$ from L2R and R2L decoding directions on the official validation set.

For Zh→Ja task, Table 2a shows that the standard Transformer-Big model outperforms all the Transformer models with deeper layers or wider dimension by a small margin and achieves the best BLEU score for L2R direction. For R2L direction, however, the deeper Transformer model with 12 layers in both the encoder and the decoder provides a significant improvement as compared to the Transformer-Big model and obtains the best BLEU score.

For Ja→Zh task, Table 2b indicates that all deep Transformer models are superior to the shallow Transformer-Big model for both the L2R and R2L directions. For L2R direction, the Transformer-Deep-12-6 model obtains the best BLEU score. For R2L direction, the Transformer-DLCL-Deep outperforms other models, particularly up to 0.44 BLEU score as compared to the Transformer-Deep-12-12. The result also demonstrates that DLCL is useful for training deep models.

For both translation tasks, although with far fewer parameters than the Transformer-Big model, the Transformer-Evolved model still obtains a competitive performance among all the baseline systems. Table 2b shows that the performance of the R2L Transformer-Evolved model ranks second among all the models for Ja→Zh. It is interesting to note that L2R decoding behaves better than that of R2L decoding, and Ja→Zh has an opposite phenomenon. We suspect that the main reason is that Chinese is a subject–verb–object (SVO) language, while Japanese is a subject–object–verb (SOV) language.

[10] http://web-language-models.s3-website-us-east-1.amazonaws.com/ngrams/zh/deduped/zh.deduped.xz

[11] http://web-language-models.s3-website-us-east-1.amazonaws.com/ngrams/ja/deduped/ja.deduped.xz

System	Constrained						Unconstrained	
	$\mathcal{D}_1$		$\mathcal{D}_2$		$\mathcal{D}_3$		$\mathcal{D}_4$	
	L2R	R2L	L2R	R2L	L2R	R2L	L2R	R2L
Transformer-Big	**37.08**	36.28	37.98	37.59	37.88	37.88	39.08	39.35
Transformer-Wide	36.98	36.57	**38.20**	37.46	38.35	38.14	39.50	39.23
Transformer-Deep-6-12	36.84	36.71	37.73	37.60	37.77	38.04	38.98	39.28
Transformer-Deep-12-6	36.96	36.50	37.58	37.07	37.70	**38.32**	38.92	39.11
Transformer-Deep-12-12	37.00	**36.97**	38.06	**37.74**	**38.39**	38.17	39.42	39.27
Transformer-DLCL-Big	36.46	36.10	37.65	37.11	38.18	37.52	39.27	**39.49**
Transformer-DLCL-Deep	36.61	36.27	37.34	37.53	37.57	37.85	**39.59**	39.17
Transformer-Evolved	36.47	35.87	37.42	36.85	37.71	37.19	38.78	38.65
Ensemble	38.37	37.96	39.22	38.82	39.32	39.20	40.1	40.13
+Reranking	-		-		$39.37^{\#}$		41.54^{*}	

(a) Chinese→Japanese

System	Constrained						Unconstrained	
	$\mathcal{D}_1$		$\mathcal{D}_2$		$\mathcal{D}_3$		$\mathcal{D}_4$	
	L2R	R2L	L2R	R2L	L2R	R2L	L2R	R2L
Transformer-Big	31.44	31.81	32.27	32.62	33.88	33.80	34.23	34.21
Transformer-Wide	31.25	31.98	32.11	32.84	33.58	34.07	34.02	34.25
Transformer-Deep-6-12	31.55	31.98	32.26	32.67	34.07	34.11	34.29	34.20
Transformer-Deep-12-6	**31.96**	32.22	32.15	32.94	33.59	34.07	33.98	**34.35**
Transformer-Deep-12-12	31.95	32.22	**32.49**	32.75	**34.22**	**34.15**	**34.31**	34.30
Transformer-DLCL-Big	31.82	32.07	32.31	32.64	34.09	34.03	34.29	34.18
Transformer-DLCL-Deep	31.64	**32.66**	32.46	33.18	34.11	34.12	34.17	34.21
Transformer-Evolved	31.44	32.46	31.99	**33.45**	32.95	33.91	33.97	33.99
Ensemble	32.57	33.17	33.30	33.79	34.73	34.52	34.82	34.85
+Reranking	-		-		$34.78^{\#}$		34.91^{*}	

(b) Japanese→Chinese

Table 2: Results of various system trained on different training data evaluated on the Chinese↔Japanese validation sets. $\mathcal{D}_1$ (16M sentence pairs) is the starting training data. $\mathcal{D}_2$ (32M sentence pairs) is $\mathcal{D}_1$ concatenated with the pseudo-parallel data back-translated from the target side of $\mathcal{D}_1$ by the ensemble models based on the T2S single models trained on $\mathcal{D}_1$. $\mathcal{D}_3$ (64M sentence pairs) is $\mathcal{D}_2$ concatenated with two KD synthetic data, including translating the source side of $\mathcal{D}_1$ by the ensemble models from the S2T-L2R single models and the ensemble models from the S2T-R2L single models that are both trained on $\mathcal{D}_2$. Finally, for one S2T language pair, the external target monolingual data is translated by the T2S-L2R Transformer-Big model trained on $\mathcal{D}_3$. The generated synthetic corpus is splitted into eight parts equally. Each part (20M sentence pairs) is concatenated with $\mathcal{D}_3$ to generate the training data $\mathcal{D}_4$ (84M sentence pairs) that is applied to train one of all the eight baseline systems. For the given decoding direction and training data, result of the best single system is bold-faced. * denotes the submitted primary system in the unconstrained condition where only the provided training data is used. $^{\#}$ denotes the submitted contrastive system in the constrained condition where external public monolingual data is applied.

4.4 Results of Systems Features

In constrained condition, Table 2 shows that BT based on the target of bilingual data also brings large improvement to all baseline systems for both the Zh→Ja and Ja→Zh tasks. We observe a solid improvement of an average BLEU score of 0.95 for Zh→Ja and an average BLEU score of 0.67 for Ja→Zh. It is worth noting that the Transformer-Evolved model achieves the best BLEU score among all the R2L systems for Ja→Zh. The result suggests that the human-designed architectures may not be optimal. Therefore, it seems promising to replace the manual process of architecture design with NAS.

Table 2a shows that the improvement of KD is relatively slight for Zh→Ja. However, the translation quality of Ja→Zh strong models after BT is further largely improved using KD, up to an average BLEU score of 0.89. We attribute this finding to the quality gap between the provided Chinese and Japanese data.

Table 2a shows that adding large-scale synthetic parallel data back-translated from external monolingual data further boost the performance in different degree. Both the best baseline systems obtain a significant improvement by 1.32 1.32 BLEU score for Zh→Ja. However, it is currently not clear to us how to interpret on the marginal improvement for Ja→Zh. There is a reason to conjecture that we might be suffering from reference bias towards translationese and non-native data (Toral et al., 2018).

Unsurprisingly, utilizing diverse models with homogeneous architectures to the ensemble improves translation quality across both the tasks in different degrees. In constrained condition, the Zh→Ja ensemble models gain a substantial improvement compared to the baseline

From the Table 2a, our reranking model finally achieves a significant improvement of about 1.4 BLEU score for Zh→Ja, even when applied on top of an ensemble of very strong KD+BT models. However, the improvement of reranking is relatively inconsiderable for Ja→Zh, and we also attribute this to the issue of translationese reference above.

5 Conclusions

We present the Xiaomi's NMT systems for IWSLT 2020 Chinese↔Japanese open domain translation tasks. For both translation tasks, our final systems achieved substantial improvements up by about 9 BLEU score over baseline systems by integrating careful data filtering, data augmentation, and other effective NMT techniques. As a result, our submitted Chinese→Japanese system rank second to the official evaluation set in terms of character-level BLEU and Japanese→Chinese system also achieves a competitive performance.

References

Ebrahim Ansari, Amittai Axelrod, Nguyen Bach, Ondrej Bojar, Roldano Cattoni, Fahim Dalvi, Nadir Durrani, Marcello Federico, Christian Federmann, Jiatao Gu, Fei Huang, Kevin Knight, Xutai Ma, Ajay Nagesh, Matteo Negri, Jan Niehues, Juan Pino, Elizabeth Salesky, Xing Shi, Sebastian Stüker, Marco Turchi, and Changhan Wang. 2020. Findings of the IWSLT 2020 Evaluation Campaign. In *Proceedings of the 17th International Conference on Spoken Language Translation (IWSLT 2020)*, Seattle, USA.

Mikel Artetxe and Holger Schwenk. 2019. Margin-based parallel corpus mining with multilingual sentence embeddings. In *Proc. of ACL*, pages 3197–3203.

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

Ankur Bapna, Mia Chen, Orhan Firat, Yuan Cao, and Yonghui Wu. 2018. Training deeper neural machine translation models with transparent attention. In *Proc. of EMNLP*, pages 3028–3033.

Marine Carpuat, Yogarshi Vyas, and Xing Niu. 2017. Detecting cross-lingual semantic divergence for neural machine translation. In *Proc. of the First Workshop on Neural Machine Translation*, pages 69–79.

Isaac Caswell, Ciprian Chelba, and David Grangier. 2019. Tagged back-translation. In *Proc. of the Fourth Conference on Machine Translation*, pages 53–63.

Yun Chen, Yang Liu, Yong Cheng, and Victor O K Li. 2017. A teacher-student framework for zero-resource neural machine translation. In *Proc. of ACL*, pages 1925–1935.

Alexis Conneau and Guillaume Lample. 2019. Cross-lingual language model pretraining. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems 32*, pages 7059–7069. Curran Associates, Inc.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proc. of NAACL*, pages 4171–4186.

Chris Dyer, Victor Chahuneau, and Noah A. Smith. 2013. A simple, fast, and effective reparameterization of IBM model 2. In *Proc. of NAACL*, pages 644–648.

Thomas Elsken, Jan Hendrik Metzen, and Frank Hutter. 2018. Neural architecture search: A survey. *arXiv preprint arXiv:1808.05377*.

Markus Freitag, Yaser Al-Onaizan, and Baskaran Sankaran. 2017. Ensemble distillation for neural machine translation. *arXiv preprint arXiv:1702.01802*.

Tommaso Furlanello, Zachary Lipton, Michael Tschannen, Laurent Itti, and Anima Anandkumar. 2018. Born again neural networks. In *Proc. of ICML*, pages 1607–1616.

Jiatao Gu, James Bradbury, Caiming Xiong, Victor O K Li, and Richard Socher. 2018. Non-autoregressive neural machine translation. In *Proc. of ICLR*.

Hany Hassan, Anthony Aue, Chang Chen, Vishal Chowdhary, Jonathan Clark, Christian Federmann, Xuedong Huang, Marcin Junczys-Dowmunt, William Lewis, Mu Li, et al. 2018. Achieving human parity on automatic chinese to english news translation. *arXiv preprint arXiv:1803.05567*.

Kenneth Heafield. 2011. Kenlm: Faster and smaller language model queries. In *Proc. of the Sixth Workshop on Statistical Machine Translation*, pages 187–197.

Geoffrey Hinton, Oriol Vinyals, and Jeffrey Dean. 2015. Distilling the knowledge in a neural network. In *NIPS Deep Learning and Representation Learning Workshop*.

Thorsten Joachims. 2006. Training linear svms in linear time. In *Proc. of SIGKDD*, pages 217–226.

Huda Khayrallah and Philipp Koehn. 2018. On the impact of various types of noise on neural machine translation. In *Proc. of the 2nd Workshop on Neural Machine Translation and Generation*, pages 74–83.

Yoon Kim and Alexander M Rush. 2016. Sequence-level knowledge distillation. In *Proc. of EMNLP*, pages 1317–1327.

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

Reinhard Kneser and Hermann Ney. 1995. Improved backing-off for m-gram language modeling. In *Proc. of ICASSP*, pages 181–184.

Philipp Koehn, Huda Khayrallah, Kenneth Heafield, and Mikel L Forcada. 2018. Findings of the wmt 2018 shared task on parallel corpus filtering. In *Proc. of the Third Conference on Machine Translation*, pages 726–739.

Taku Kudo. 2006. Mecab: Yet another part-of-speech and morphological analyzer. *http://mecab.sourceforge.jp*.

Lemao Liu, Masao Utiyama, Andrew Finch, and Eiichiro Sumita. 2016. Agreement on target-bidirectional neural machine translation. In *Proc. of NAACL*, pages 411–416.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Phoebe Mulcaire, Jungo Kasai, and Noah A Smith. 2019. Polyglot contextual representations improve crosslingual transfer. In *Proc. of NAACL*, pages 3912–3918.

Sharan Narang, Gregory Diamos, Erich Elsen, Paulius Micikevicius, Jonah Alben, David Garcia, Boris Ginsburg, Michael Houston, Oleksii Kuchaiev, Ganesh Venkatesh, et al. 2018. Mixed precision training. In *Proc. of ICLR*.

Nathan Ng, Kyra Yee, Alexei Baevski, Myle Ott, Michael Auli, and Sergey Edunov. 2019. Facebook FAIR's WMT19 news translation task submission. In *Proceedings of the Fourth Conference on Machine Translation*, pages 314–319.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proc. of ACL*, pages 311–318.

Marc'Aurelio Ranzato, Sumit Chopra, Michael Auli, and Wojciech Zaremba. 2016. Sequence level training with recurrent neural networks. In *Proc. of ICLR*.

Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence embeddings using Siamese BERT-networks. In *Proc. of EMNLP*, pages 3982–3992.

Holger Schwenk. 2018. Filtering and mining parallel data in a joint multilingual space. In *Proc. of ACL*, pages 228–234.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016a. Edinburgh neural machine translation systems for WMT 16. In *Proc. of the First Conference on Machine Translation*, pages 371–376.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016b. Improving neural machine translation models with monolingual data. In *Proc. of ACL*, pages 86–96.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016c. Neural machine translation of rare words with subword units. In *Proc. of ACL*, pages 1715–1725.

Noam Shazeer and Mitchell Stern. 2018. Adafactor: Adaptive learning rates with sublinear memory cost. *arXiv preprint arXiv:1804.04235*.

Libin Shen, Anoop Sarkar, and Franz Josef Och. 2004. Discriminative reranking for machine translation. In *Proc. of NAACL*, pages 177–184.

David R So, Chen Liang, and Quoc V Le. 2019. The evolved transformer. In *Proc. of ICML*.

Miloš Stanojević and Khalil Sima'an. 2014. BEER: BEtter evaluation as ranking. In *Proc. of the Ninth Workshop on Statistical Machine Translation*, pages 414–419.

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

Xu Tan, Yi Ren, Di He, Tao Qin, Zhou Zhao, and Tie-Yan Liu. 2019. Multilingual neural machine translation with knowledge distillation. In *Proc. of ICLR*.

Antonio Toral, Sheila Castilho, Ke Hu, and Andy Way. 2018. Attaining the unattainable? reassessing claims of human parity in neural machine translation. In *Proc. of the Third Conference on Machine Translation: Research Papers*, pages 113–123.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems 30*, pages 5998–6008. Curran Associates, Inc.

Qiang Wang, Bei Li, Tong Xiao, Jingbo Zhu, Changliang Li, Derek F Wong, and Lidia S Chao. 2019. Learning deep transformer models for machine translation. In *Proc. of ACL*, pages 1810–1822.

Wei Wang, Taro Watanabe, Macduff Hughes, Tetsuji Nakagawa, and Ciprian Chelba. 2018. Denoising neural machine translation training with trusted data and online data selection. In *Proc. of WMT*.

Yuguang Wang, Xiang Li, Shanbo Cheng, Liyang Jiang, Jiajun Yang, Wei Chen, Lin Shi, Yanfeng Wang, and Hongtao Yang. 2017. Sogou neural machine translation systems for WMT17. In *Proc. of the Second Conference on Machine Translation*, pages 410–415.

Lijun Wu, Yiren Wang, Yingce Xia, Fei Tian, Fei Gao, Tao Qin, Jianhuang Lai, and Tie-Yan Liu. 2019. Depth growing for neural machine translation. In *Proc. of ACL*, pages 5558–5563.

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Russ R Salakhutdinov, and Quoc V Le. 2019. Xlnet: Generalized autoregressive pretraining for language understanding. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems 32*, pages 5753–5763. Curran Associates, Inc.

Biao Zhang, Ivan Titov, and Rico Sennrich. 2019. Improving deep transformer with depth-scaled initialization and merged attention. In *Proc. of EMNLP*, pages 897–908.

Xiangwen Zhang, Jinsong Su, Yue Qin, Yang Liu, Rongrong Ji, and Hongji Wang. 2018. Asynchronous bidirectional decoding for neural machine translation. In *Proc.s of AAAI*.

Long Zhou, Jiajun Zhang, and Chengqing Zong. 2019. Synchronous bidirectional neural machine translation. *Transactions of the Association for Computational Linguistics*, 7:91–105.