NICT-5's Submission To WAT 2021: MBART Pre-training And In-Domain Fine Tuning For Indic Languages

Raj Dabre[‡] **Abhisek Chakrabarty**[‡]
[‡]National Institute of Information and Communications Technology, Kyoto, Japan
{raj.dabre, abhisek.chakra}@nict.go.jp

Abstract

In this paper we describe our submission to the multilingual Indic language translation task "MultiIndicMT" under the team name "NICT-5". This task involves translation from 10 Indic languages into English and vice-versa. The objective of the task was to explore the utility of multilingual approaches using a variety of in-domain and out-of-domain parallel and monolingual corpora. Given the recent success of multilingual NMT pre-training we decided to explore pre-training an MBART model on a large monolingual corpus collection covering all languages in this task followed by multilingual fine-tuning on small in-domain corpora. Firstly, we observed that a small amount of pre-training followed by fine-tuning on small bilingual corpora can yield large gains over when pre-training is not used. Furthermore, multilingual fine-tuning leads to further gains in translation quality which significantly outperforms a very strong multilingual baseline that does not rely on any pre-training.

1 Introduction

Neural machine translation (NMT) (Bahdanau et al., 2014) is known to give state-of-the-art translations for a variety of language pairs. NMT is known to perform poorly for language pairs for which parallel corpora are scarce. This happens due to lack of translation knowledge as well as due to overfitting which is inevitable in a low-resource setting. Fortunately, transfer learning via cross-lingual transfer (Zoph et al., 2016; Dabre et al., 2019), multilingualism (Firat et al., 2016; Dabre et al., 2020), back-translation (Sennrich et al., 2016) or monolingual pre-training (Liu et al., 2020; Lewis et al., 2020; Mao et al., 2020) can significantly improve translation quality in a low-resource situation.

Cross-lingual transfer learning involves pre-training a model using a parallel corpus for a resource-rich language pair $XX - YY$ and then fine-tuning on a parallel corpus for a resource-poor language pair $AA - BB$. Naturally the improvements in translation quality will be impacted by if $XX = AA$ or $YY = BB$[1] and it is often better to have a shared target language. Cross-lingual transfer despite its simplicity and effectiveness relies on shared source or target languages for effective transfer and thus depending on methods that use monolingual corpora are preferable. This also applies to vanilla multilingual training which does not rely on monolingual corpora. Another reason for focusing on utilizing monolingual corpora is that they are extremely abundant when compared to parallel corpora and they contain a large amount of language modeling information. In this regard, back-translation and multilingual pre-training are two of the most reliable methods.

While back-translation is easy to use, it involves the translation of millions of monolingual sentences and quite often it is necessary to perform multiple iterations of the back-translation process to yield the best results (Hoang et al., 2018) which means that it is quite resource intensive. This leaves us with multilingual pre-training using methods such as BART/MBART (Liu et al., 2020; Lewis et al., 2020) which we use for developing our translation system. The advantage of BART/MBART is that we need to pre-train these models once and then fine-tune not only for machine translation but also for any natural language generation task such as summarization (Shi et al., 2021). These models can be upgraded to include additional language pairs in the future by simply resuming pre-training (Tang et al., 2020).

In this paper, we describe our simple approach involving MBART pre-training and fine-tuning. First, we use the official monolingual corpora to train an MBART model spanning all 11 languages in

[1]If $XX - YY$ and $AA - BB$ are the same pairs then it is known as domain adaptation.

Proceedings of the 8th Workshop on Asian Translation, pages 198–204

the shared task. Following this we fine-tune the MBART model using the officially provided in-domain corpora in two different ways: bilingual fine-tuning and multilingual fine-tuning. Additionally we also train multilingual models without any pre-training. The multilingual models are one-to-many (English to Indic) and many-to-one (Indic to English) in nature. The bilingual fine-tuning and non pre-trained multilingual model serve as strong baselines which significantly outperform the organizers weak bilingual baselines. Our multilingual fine-tuning models exhibit the best translation quality out of all our models which shows the power of effectively combining monolingual corpora with multilingualism.

We refer readers to the workshop overview paper (Nakazawa et al., 2021) for a better understanding of the task and the comparison of our results with those of other participants.

2 Related Work

The techniques used in this paper revolve around multilingualism, sequence-to-sequence pre-training and transfer learning.

Firat et al. (2016) proposed multilingual neural translation using multiple encoders and decoders which was then simplified by Johnson et al. (2017) to require a single encoder and decoder to be shared among multiple language pairs. Due to the simplicity of the latter approach, most modern multilingual models are based on it and in this paper we also use the same approach. Multilingualism involves implicit transfer learning but a more explicit way to do the same is to use fine-tuning (Zoph et al., 2016). However all these aforementioned approaches rely on bilingual data which is not always readily available. This can be remedied by the use of monolingual corpora for backtranslation (Sennrich et al., 2016) or for pre-training (Lewis et al., 2020; Liu et al., 2020; Mao et al., 2020). As backtranslation is resource intensive, given that it involves translation of a large amount of monolingual corpora, pre-training is more attractive as a pre-trained model can be used for a variety of natural language generation tasks. In this paper we combine sequence-to-sequence pre-training followed by multilingual fine-tuning. For an overview of multilingual NMT we refer readers to a survey paper on multilingualism and low-resource NMT in general (Dabre et al., 2020).

3 Our Approaches

For our submissions we focused on combining multilingual denoising pre-training (MBART) and multilingual fine tuning.

3.1 Multilingual NMT Training

We follow the multilingual NMT training approach proposed by Johnson et al. (2017). Consider a multilingual parallel corpora collection spanning corpora for N language pairs $L^i_{src} - L^i_{tgt}$ for $i \in [1, N]$. The sizes of the parallel corpora are typically different, often radically different, in which case it is important to balance corpora sizes to prevent the model from focusing too much on some language pairs. Johnson et al. (2017) showed that training by oversampling smaller corpora to match the size of the largest corpus is the best approach. However, since then newer corpora balancing approaches have been proposed and the most recent effective method is known as the temperature based sampling approach (Aharoni et al., 2019). Suppose that the size of the i^{th} corpus is s_i which means the probability of sampling a sentence pair from each corpus is $p_i = \frac{s_i}{S}$ where $S = \sum_i s_i$. Using this default sampling probability is biased towards larger corpora so first the probability values are tempered using a temperature T. The resultant probabilities p^t_i are obtained as follows:

$$p^t_i = \frac{p_i^{\frac{1}{T}}}{\sum_j p_j^{\frac{1}{T}}} \quad (1)$$

When $T = 1$, $p^t_i = p_i$ and when $T = \infty$, $p^t_i = \frac{1}{N}$. Aharoni et al. (2019) showed that a value of $T = 5$ works well in practice which is what we use in our experiments. During training, sentence pairs are sampled from each corpus following which the source sentence is prepended with a token $< 2L^i_{tgt} >$ which indicates that the source sentence should be translated into L^i_{tgt}. Thereafter, the pre-processed source sentence and target sentence are fed to the NMT model which learns how to translate between multiple language pairs.

3.2 MBART Pre-training and Fine-Tuning

Liu et al. (2020) extended the BART model (Lewis et al., 2020) by denoising pre-training the BART model on 25 languages instead of 2 which leads to an MBART model. The main advantage of an MBART model is that it can be fine-tuned with corpora for a variety of language pairs which naturally

Language	#Lines
as	1.39M
bn	39.9M
en	54.3M
gu	41.1M
hi	63.1M
kn	53.3M
ml	50.2M
mr	34.0M
or	6.94M
pa	29.2M
ta	31.5M
te	47.9M

Table 1: Monolingual corpora statistics.

Language Pair	#Lines
bn-en	23,306
gu-en	41,578
hi-en	50,349
kn-en	28,901
ml-en	26,916
mr-en	28,974
or-en	31,966
pa-en	28,294
ta-en	32,638
te-en	33,380

Table 2: Bilingual corpora statistics for the PMI dataset only.

includes many zero-shot pairs. The way to train an MBART model is by "corrupting" an input sentence, feeding it to the encoder and then training the model to predict the original sentence. Corruption can be done in a variety of ways and in this paper we use 'text infilling' approach which finds random spans of the source tokens and replaces them with a token such as $< MASK >$ till a certain percentage of the sentence is masked. The length of the span is sampled from a Poisson distribution with a mean of λ. Liu et al. (2020) determined an optimal value of $\lambda = 3.5$ which we also use. The denoising objective helps the MBART model learn about using context to translate and also helps it acquire language modeling information.

After an MBART model is trained it is fine-tuned on a bilingual or multilingual parallel corpus which is then used for translation. The language modeling priors help account for missing translation knowledge in low-resource settings which leads to large improvements in translation quality over baselines which only use parallel corpora.

4 Experimental Setup

Our goal was to study how far the translation quality can be pushed via MBART pre-training and multilingual fine-tuning. To do so, we describe the datasets, implementation details, evaluation metrics and the models trained.

4.1 Datasets and Preprocessing

The languages involved in the task are: Bengali, Gujarati, Hindi, Kannada, Malayalam, Marathi, Oriya, Punjabi, Tamil, Telugu and English. We used the official parallel corpora[2] provided by the organizers. The 11-way evaluation development and test sets come from the PMI dataset[3]. Although the organizers provided corpora from other sources as well, we decided to restrict ourselves to the PMI part of the parallel corpora to avoid the need for data selection. Instead we relied on pre-training to compensate for using smaller amount of parallel corpora. For MBART pre-training we used the AI4Bharat's monolingual corpora known as IndicCorp [4] (Kunchukuttan et al., 2020). Note that MBART pre-training supposes the monolingual data is available as documents however since we only use the masking denoising approach, sentence level corpora[5] are sufficient. The IndicCorp covers an additional language Assamese which is not in this shared task. Nevertheless, we use the monolingual corpus for this language as well because it can potentially improve translation involving Bengali given their similarity. However, the small size of Assamese data (1.39M lines) relative to the Bengali data (39.9M lines) should not significantly affect the final outcome for translation involving Benglai[6]. The monolingual corpora stats are given in Table 1 and the bilingual corpora stats are given in Table 2.

Regarding pre-processing, we do not perform anything specific and instead let our implementation handle everything via its internal mechanisms.

[2] https://lotus.kuee.kyoto-u.ac.jp/WAT/indic-multilingual/index.html

[3] http://data.statmt.org/pmindia

[4] https://indicnlp.ai4bharat.org/corpora

[5] The IndicCorp is supposed to be document level but the downloadable version is sentence level.

[6] However, this may significantly improve translation involving Assamese thanks to the Bengali data.

4.2 Implementation Details

We implement the methods mentioned in Section 3 in our in-house toolkit which we make publicly available[7]. This toolkit is based on the HuggingFace transformers library (Wolf et al., 2020) v4.3.2. Note that the MBART implementation in the library shares the encoder embedding, decoder embedding and decoder softmax projection layers. We implement denoising, temperature based data sampling and multilingual training ourselves. We also use the HuggingFace transformer tokenizer library to train tokenizers. These tokenizers are wrappers around Byte Pair Encoding (BPE) (Gage, 1994) or SentencePiece (SPM) (Kudo and Richardson, 2018) models and we choose[8] the latter as opposed to the former which is used by the original MBART implementation.

4.3 Training and Evaluation

We first trained a tokenizer with a joint vocabulary size of 64,000 sub-words which is learned on the IndicCorp monolingual data. We consider this vocabulary size to be sufficient for all languages. For pre-training, we use hyperparameters corresponding to the "transformer_big" (Vaswani et al., 2017) with a few exceptions such as dropout of 0.1, positional embeddings instead of positional encodings and a maximum learning rate of 0.001. When performing batching we truncate all sequences longer than 256 subwords. Our MBART model is pre-trained on 48 NVIDIA V-100 GPUs using the distributed data parallel mechanism in PyTorch. Due to lack of time we only trained for 150,000 batches which corresponded to roughly 1 epoch over the entire monolingual data. After pre-training we train unidirectional models using the bilingual data on a single GPU. We train the one-to-many (English to Indic) and many-to-one (Indic to English) models on the multilingual data on 8 GPUs. For both cases we use a dropout of 0.3 and train till convergence on the development BLEU score and choose the model with the best development set BLEU score for decoding the test set. In our initial experiments we did additional exploration to choose the particular checkpoint which yields best average development BLEU score over all language pairs for decoding the test set. We found that the results are inferior compared to when the best model is chosen language pairwise. We use beam search for decoding with a beam size of 4 and a length penalty of 0.8[9]. For unidirectional models this is strightforward but for multilingual models train till convergence on the global development set BLEU score, an average of BLEU scores for each language pair. Different from most previous works, instead of decoding a single final model, we choose a particular model for a language pair with the highest development set BLEU score for that pair. Therefore, we treat multilingualism as a way get a (potentially) different model per language pair leading to the best BLEU scores for that pair and not as a way to get a single model that gives the best performance for each language pair.

For evaluation, as we have mentioned before, we use BLEU (Papineni et al., 2002) as the primary evaluation metric. WAT also uses metrics such as RIBES (Isozaki et al., 2010), AM-FM (Zhang et al., 2021) and human evaluation (Nakazawa et al., 2019, 2020, 2021). All these metrics focus on different aspects of translations and may lead to different rankings for submissions, however this multi-metric evaluation helps us understand that there may not be one perfect model. To avoid confusing the reader with a clutter of scores, we only show BLEU scores and we refer the reader to the evaluation page where all scores and rankings[10] can be seen[11].

4.4 Models Trained

We trained the following models:

- A pre-trained MBART model.
- Unidirectional models for each language pair trained from scratch or via fine-tuning the MBART model.
- One-to-many (English to Indic) and many-to-one (Indic to English) multilingual models trained from scratch or via fine-tuning the MBART model.

[7] https://github.com/prajdabre/yanmtt

[8] We choose SPM because SPM can work with unsegmented, untokenized raw text for any language. Inside the transformers library, the AlbertTokenizer acts as a wrapper for the SPM model. Our implementation also allows the usage of the BPE model but we do not use it in this paper.

[9] We have not tuned these decoding hyperparameters and our BLEU scores may improve.

[10] As can be seen, the rankings of translation can change depending on the metric which indicates that multi-metric ranking is important

[11] http://lotus.kuee.kyoto-u.ac.jp/WAT/evaluation/index.html

Model	Source Language									
	Bn	Gu	Hi	Kn	Ml	Mr	Or	Pa	Ta	Te
Unidirectional	11.27	26.21	28.21	20.33	13.64	15.10	16.35	23.66	16.07	14.70
Many-to-one	20.06	27.72	30.86	24.66	21.79	22.66	23.04	27.61	21.90	23.39
MBART+ Unidirectional	21.37	**33.65**	35.80	29.29	26.55	25.45	25.81	34.34	24.72	27.76
MBART+ Many-to-one	**23.89**	33.53	**36.20**	**30.87**	**28.23**	**27.88**	**27.93**	**35.81**	**26.90**	**28.77**
Official Best Submission	31.87	43.98	46.93	40.34	38.38	36.64	37.06	46.39	36.13	39.80
Model	Target Language									
	Bn	Gu	Hi	Kn	Ml	Mr	Or	Pa	Ta	Te
Unidirectional	5.58	16.38	23.31	10.11	3.34	8.82	9.08	21.77	6.38	2.80
One-to-many	11.56	23.49	29.12	17.53	6.22	15.01	16.43	28.37	10.82	3.81
MBART+ Unidirectional	10.59	23.04	29.59	16.13	5.98	14.69	15.01	26.94	10.33	**4.59**
MBART+ One-to-many	**12.84**	**24.26**	**30.18**	**18.22**	**6.51**	**16.38**	**16.69**	**29.15**	**11.42**	4.20
Official Best Submission	15.97	27.80	38.65	21.30	15.49	20.42	20.15	33.43	14.43	16.85

Table 3: Evaluation results of all language pairs. All scores are taken from the leaderboard. Our best results are in bold. Differences in BLEU smaller than 0.5 are not significant in most cases.

5 Results and Observations

Table 3 contains the results of the unidirectional[12], and multilingual models. We also show the the best submissions for reference.

5.1 Without Fine-tuning

It is clear from the results that multilingual models are vastly superior than unidirectional models which shows that multilingualism is very helpful in a low-resource setting. Secondly, comparing with corpora sizes (see Table 2), it can be seen that the gains in BLEU are (roughly) inversely proportional to the size of the parallel corpora.

5.2 Non Fine-Tuned Multilingual Models vs Fine-Tuned Unidirectional Models

In the case of Indic to English translation, MBART+unidirectional models are significantly better than many-to-one models. We can attribute this phenomenon to the fact that the PMI corpus has a limited number of English sentences and even though combining all corpora might seem to increase the number of English sentences, most of them are redundant which causes some form of overfitting. This is remedied by the MBART model with incorporates additional language modeling information through the monolingual corpora.

On the other hand, for English to Indic translation, the one-to-many models are often comparable if not better than the fine-tuned unidirectional models. Fine-tuning significantly outperforms non fine-tuned unidirectional models which means pre-training is useful. However, given that multilingual training is better, this indicates that it may not be necessary to perform pre-training for one-to-many translation. Remember that the English side of the text contains a large number of redundant sentences and this may be one of the reasons for this kind of behavior. We think that this deserves some future investigation.

5.3 Multilingual Fine-tuning

Ultimately, multilingual fine-tuning of an MBART model leads to the best translation quality for all language pairs, except two (Gujarati to English and English to Telugu). This approach combines the best of both worlds and the outcome is not surprising. Our MBART models consisted of only 6 layers and was trained for only 1 epoch and this may not be enough to incorporate knowledge from the full monolingual corpus. We also did not perform any hyperparameter tuning with parameters such as dropout and learning rate[13] We expect that a larger model with more careful hyperparameter tuning should lead to even better results. However, we are

[12]The unidirectional scores without fine-tuning are actually organizer baselines but we were the ones who actually developed them so we use the scores as is.

[13]We used a high learning rate which may not have been a good idea in retrospect.

confident that a multilingual fine-tuned model will reign supreme.

5.4 Comparison With Other Submissions

For Indic to English translation the several submissions outperformed ours and we think that this is because the other participants have indicated that they have performed data selection, backtranslation and script mapping. In our case we only performed pre-training and fine-tuning with PMI data. Although MBART pre-training is helpful, it can never compare with the power of a large parallel corpus obtained via careful data selection and script manipulation. While for PMI, the largest parallel corpus, Hindi-English, contains roughly 50,000 lines, the full Hindi-English corpus is larger than 2M lines and most pairs have more than 500,000 lines. In the future we will try training with larger parallel corpora and script mapping to see what kind of results we get.

On the other hand for English to Indic translation, the gap between the the best submissions and ours is much smaller than for the reverse direction. This also shows that, at least for this task, multilingualism benefits translation into English a lot more than it benefits translation from English.

6 Conclusion

In this paper we have described our NMT systems and results for the MultiIndicMT task in WAT 2021. We worked on MBART pre-training and multilingual fine-tuning which we found to significantly outperform unidirectional models with and without pre-training and multilingual models without pre-training. We did not train our MBART models for more than 1 epoch and used only the PMI data for fine-tuning instead of the whole parallel corpus. We did not try any additional methods such as back-translation either. Despite this, our results are competitive and despite the simplicity of our methods our results do not lag far behind those of the best systems that use advanced methods such as data selection, domain adaptation, back-translation etc. This also means that we have a lot of room for improvement in the future.

References

Roee Aharoni, Melvin Johnson, and Orhan Firat. 2019. Massively multilingual neural machine translation. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 3874–3884, Minneapolis, Minnesota. Association for Computational Linguistics.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2014. Neural Machine Translation by Jointly Learning to Align and Translate. *arXiv e-prints*, page arXiv:1409.0473.

Raj Dabre, Chenhui Chu, and Anoop Kunchukuttan. 2020. A survey of multilingual neural machine translation. *ACM Comput. Surv.*, 53(5).

Raj Dabre, Atsushi Fujita, and Chenhui Chu. 2019. Exploiting multilingualism through multistage fine-tuning for low-resource neural machine translation. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 1410–1416, Hong Kong, China. Association for Computational Linguistics.

Orhan Firat, Kyunghyun Cho, and Yoshua Bengio. 2016. Multi-way, multilingual neural machine translation with a shared attention mechanism. In *NAACL HLT 2016, The 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, San Diego California, USA, June 12-17, 2016*, pages 866–875. The Association for Computational Linguistics.

Philip Gage. 1994. A new algorithm for data compression. *C Users J.*, 12(2):23–38.

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.

Hideki Isozaki, Tsutomu Hirao, Kevin Duh, Katsuhito Sudoh, and Hajime Tsukada. 2010. Automatic evaluation of translation quality for distant language pairs. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, pages 944–952, Cambridge, MA. Association for Computational Linguistics.

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

Taku Kudo and John Richardson. 2018. SentencePiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. In

Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations, pages 66–71, Brussels, Belgium. Association for Computational Linguistics.

Anoop Kunchukuttan, Divyanshu Kakwani, Satish Golla, Gokul N. C., Avik Bhattacharyya, Mitesh M. Khapra, and Pratyush Kumar. 2020. Ai4bharat-indicnlp corpus: Monolingual corpora and word embeddings for indic languages. *CoRR*, abs/2005.00085.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880, Online. Association for Computational Linguistics.

Yinhan Liu, Jiatao Gu, Naman Goyal, Xian Li, Sergey Edunov, Marjan Ghazvininejad, Mike Lewis, and Luke Zettlemoyer. 2020. Multilingual denoising pre-training for neural machine translation. *Transactions of the Association for Computational Linguistics*, 8:726–742.

Zhuoyuan Mao, Fabien Cromieres, Raj Dabre, Haiyue Song, and Sadao Kurohashi. 2020. Jass: Japanese-specific sequence to sequence pre-training for neural machine translation. In *Proceedings of The 12th Language Resources and Evaluation Conference*, pages 3683–3691, Marseille, France. European Language Resources Association.

Toshiaki Nakazawa, Chenchen Ding, Raj Dabre, Anoop Kunchukuttan, Nobushige Doi, Yusuke Oda, Ondřej Bojar, Shantipriya Parida, Isao Goto, and Hidaya Mino, editors. 2019. *Proceedings of the 6th Workshop on Asian Translation*. Association for Computational Linguistics, Hong Kong, China.

Toshiaki Nakazawa, Hideki Nakayama, Chenchen Ding, Raj Dabre, Shohei Higashiyama, Hideya Mino, Isao Goto, Win Pa Pa, Anoop Kunchukuttan, Shantipriya Parida, Ondřej Bojar, Chenhui Chu, Akiko Eriguchi, Kaori Abe, and Sadao Oda, Yusuke Kurohashi. 2021. Overview of the 8th workshop on Asian translation. In *Proceedings of the 8th Workshop on Asian Translation*, Bangkok, Thailand. Association for Computational Linguistics.

Toshiaki Nakazawa, Hideki Nakayama, Chenchen Ding, Raj Dabre, Shohei Higashiyama, Hideya Mino, Isao Goto, Win Pa Pa, Anoop Kunchukuttan, Shantipriya Parida, Ondřej Bojar, and Sadao Kurohashi. 2020. Overview of the 7th workshop on Asian translation. In *Proceedings of the 7th Workshop on Asian Translation*, pages 1–44, Suzhou, China. Association for Computational Linguistics.

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 of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics.

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.

Tian Shi, Yaser Keneshloo, Naren Ramakrishnan, and Chandan K. Reddy. 2021. Neural abstractive text summarization with sequence-to-sequence models. *ACM/IMS Trans. Data Sci.*, 2(1).

Yuqing Tang, Chau Tran, Xian Li, Peng-Jen Chen, Naman Goyal, Vishrav Chaudhary, Jiatao Gu, and Angela Fan. 2020. Multilingual translation with extensible multilingual pretraining and finetuning. *CoRR*, abs/2008.00401.

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.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.

Chen Zhang, Luis Fernando D'Haro, Rafael E. Banchs, Thomas Friedrichs, and Haizhou Li. 2021. *Deep AM-FM: Toolkit for Automatic Dialogue Evaluation*, pages 53–69. Springer Singapore, Singapore.

Barret Zoph, Deniz Yuret, Jonathan May, and Kevin Knight. 2016. Transfer learning for low-resource neural machine translation. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1568–1575, Austin, Texas. Association for Computational Linguistics.