NICT's participation to WAT 2019: Multilingualism and Multi-step Fine-Tuning for Low Resource NMT

Raj Dabre   **Eiichiro Sumita**
National Institute of Information and Communications Technology
3-5 Hikaridai, Seika-cho, Soraku-gun, Kyoto, 619-0289, Japan
firstname.lastname@nict.go.jp

Abstract

In this paper we describe our submissions to WAT 2019 for the following tasks: English–Tamil translation and Russian–Japanese translation. Our team,"NICT-5", focused on multilingual domain adaptation and back-translation for Russian–Japanese translation and on simple fine-tuning for English–Tamil translation . We noted that multi-stage fine tuning is essential in leveraging the power of multilingualism for an extremely low-resource language like Russian–Japanese. Furthermore, we can improve the performance of such a low-resource language pair by exploiting a small but in-domain monolingual corpus via back-translation. We managed to obtain second rank in both tasks for all translation directions.

1 Introduction

Neural machine translation (NMT) (Cho et al., 2014; Sutskever et al., 2014; Bahdanau et al., 2015) has enabled end-to-end training of a translation system without needing to deal with word alignments, translation rules, and complicated decoding algorithms, which are the characteristics of phrase-based statistical machine translation (PBSMT) (Koehn et al., 2007). Although vanilla NMT is significantly better than PBSMT in resource-rich scenarios, PBSMT performs better in resource-poor scenarios (Zoph et al., 2016). By exploiting transfer learning techniques, the performance of NMT approaches can be improved substantially.

For WAT 2019, we participated as team "NICT-5" and worked on Russian–Japanese and English–Tamil translation. The techniques we focused on for each translation task can be summarized as below:

- For the Russian–Japanese translation task, we submitted the results of our work presented in Imankulova et al. (2019) where we focused on multilingual stage-wise tuning followed by back-translation.
- For the English–Tamil translation task, we observed that simply fine-tuning a Hindi–English model is enough to give high quality translations.

For additional details of how our submissions are ranked relative to the submissions of other WAT participants, kindly refer to the overview paper (Nakazawa et al., 2019).

2 NMT Models and Approaches

We will first describe the Transformer which is the state-of-the-art NMT model we used for our experiments.

2.1 The Transformer

The Transformer (Vaswani et al., 2017) is the current state-of-the-art model for NMT. It is a sequence-to-sequence neural model that consists of two components, the *encoder* and the *decoder*. The encoder converts the input word sequence into a sequence of vectors of high dimensionality. The decoder, on the other hand, produces the target word sequence by predicting the words using a combination of the previously predicted word and relevant parts of the input sequence representations. Due to lack of space, we briefly describe the encoder and decoder as follows. The reader is encouraged to read the Transformer paper (Vaswani et al., 2017) for a deeper understanding.

2.2 Fine-Tuning for Transfer Learning

We use fine-tuning for transfer learning. Zoph et al. (2016) proposed to train a robust L3→L1 parent model using a large L3–L1 parallel corpus and then fine-tune it on a small L2–L1 corpus to

Proceedings of the 6th Workshop on Asian Translation, pages 76–80
Hong Kong, China, November 4, 2019. ©2019 Association for Computational Linguistics

obtain a robust L2→L1 child model. The underlying assumption is that the pre-trained L3→L1 model contains prior probabilities for translation into L1. The prior information is divided into two parts: language modeling information (strong prior) and cross-lingual information (weak or strong depending on the relationship between L3 and L2). Dabre et al. (2017) have shown that linguistically similar L3 and L2 allow for better transfer learning. As such, we used Hindi as the helping language, L3 for which L2 is Tamil because both are Indian languages. In theory, Tamil should benefit more from Dravidian languages but there is no large helping corpus involving a Dravidian language.

It is reasonable to expect improvements in translation by fine-tuning a L3→L1 model on L2→L1 data because of the additional target language monolingual data that helps improve the decoder-side language model. However, previous research has shown that this works even if the translation direction is reversed (Kocmi and Bojar, 2018). As such, we also experiment with fine-tuning a L1→L3 model on L1→L2 data with the expectation that the encoder representations will be improved.

2.3 Multilingual Multi-stage Training with Back-translation

In Imankulova et al. (2019), we proposed leveraging multilingualism via multiple training stages. Although we explain the idea in detail below, we urge the readers to read this paper for minute details regarding implementation and data-preprocessing.

Assume that our language pair of interest is L1–L2 for which we have very little data. We have the following types of helping data: large L1–L3 and L2–L3 out-of-domain parallel corpora, small L1–L3 and L2–L3 in-domain parallel corpora and in-domain monolingual corpora that are slightly larger than the in-domain parallel corpora. In order to train robust NMT models we do the following:

1. Train a multilingual L1↔L3 and L2↔L3 model using the out-of-domain data.

2. Perform domain-adaptation by fine-tuning the previous model on in-domain and out-of-domain L1↔L3 and L2↔L3 data.

3. Introduce L1–L2 pair by fine-tuning the previous model on in-domain L1↔L2, L1↔L3 and L2↔L3 data.

4. Use robust multilingual model for back-translation and final model training:

   (a) Use the previous model to back-translate all in-domain monolingual corpora for L1, L2 and L3 into all other languages.

   (b) a. Train a multilingual model for L1↔L2, L1↔L3 and L2↔L3 using all in-domain parallel and pseudo-parallel corpora.

5. Repeat N^1 times:

   (a) Use the previous model to back-translate all in-domain monolingual corpora for L1, L2 and L3 into all other languages.

   (b) a. Fine-tune the previous model using all in-domain parallel and pseudo-parallel corpora.

This stage-wise division of training ensures that the model focuses on a specific domain per training step and relies on multilingualism to address the scarcity of data. The resultant model used for back-translation leads to an inflation in good quality in-domain data which should substantially increase translation performance. In our experiments, L1 is Russian, L2 is Japanese and L3 is English.

3 Model Training Details

For all our experiments, we used the tensor2tensor[2] version 1.6 implementation of the Transformer (Vaswani et al., 2017) model. We chose this implementation because it is known to give the state-of-the-art results for NMT. For Russian–Japanese, we use the same pre/post-processing steps as mentioned in Imankulova et al. (2019). Specifically, we processed the Russian and English text using the tokenizer[3] and detokenizer[4] in Moses. We tokenized the Japanese

[1] In practice we noticed that the performance stagnates after repeating this process 3 times

[2] https://github.com/tensorflow/tensor2tensor

[3] https://github.com/moses-smt/mosesdecoder/blob/master/scripts/tokenizer/tokenizer.perl

[4] https://github.com/moses-smt/mosesdecoder/blob/master/scripts/tokenizer/detokenizer.perl

Lang.pair	Partition	#sent.	#tokens	#types
Ja↔Ru	train	12,356	341k / 229k	22k / 42k
	development	486	16k / 11k	2.9k / 4.3k
	test	600	22k / 15k	3.5k / 5.6k
Ja↔En	train	47,082	1.27M / 1.01M	48k / 55k
	development	589	21k / 16k	3.5k / 3.8k
	test	600	22k / 17k	3.5k / 3.8k
Ru↔En	train	82,072	1.61M / 1.83M	144k / 74k
	development	313	7.8k / 8.4k	3.2k / 2.3k
	test	600	15k / 17k	5.6k / 3.8k

Table 1: Statistics on our in-domain parallel data for the Russian–Japanese task.

text using Mecab[5]. Note that the implementation we used for our experiments learns and performs sub-word segmentation on the tokenized text. In order to compute BLEU we unsub-worded and detokenized Russian translations whereas we only unsub-worded Japanese translations. For Tamil–English we do not perform any specific pre/post-processing like we did for Russian–Japanese. In order to train multilingual models we used the artificial token trick used for zero-shot NMT (Johnson et al., 2017). In order to avoid vocabulary mismatches during fine-tuning we use multilingual vocabularies learned from the concatenation of all data available for a particular task. We always oversample the smaller datasets to ensure that the training phase sees equal amounts of data from all datasets. We used the default hyperparameters in tensor2tensor for all our models with the exception of the number of training iterations. We use the "base" transformer model hyperparameter settings with a 32000 subword vocabulary which is learned using tensor2tensor's default subword segmentation mechanism. During training, a model checkpoint is saved every 1000 iterations. We train models till convergence of the development set. In our implementation we used the following setting: a model is said to convergence when the BLEU score does not vary by more than 0.1 BLEU for 20,000 iterations. We averaged the last 10 model checkpoints and used it for decoding the test sets.

4 Russian↔Japanese Task

We observed that Russian↔Japanese translation shows best performance when multilingual multi-stage training is performed in conjunction with back-translation.

4.1 Datasets

For the Russian↔Japanese task tasks we used the official data provided by the organizers. Refer to Table 1 for an overview of the in-domain parallel corpora and the data splits. In addition we used out-of-domain corpora involving Russian↔English and English↔Japanese and in-domain monolingual corpora for all 3 languages. All data used was the same as in Imankulova et al. (2019). The testing domain was News Commentary and hence is challenging given the scarce amount of in-domain data.

4.2 Results

For Japanese→Russian our submission had a BLEU score of 8.11 which is substantially lower than the best system's BLEU of 14.36. On the other hand, for Russian→Japanese our submission had a BLEU score of 12.09 (JUMAN segmentation) which is not that far from the best system whose BLEU score was 15.29. For both directions, we are much better than the organizer baseline which have BLEU scores of 0.69 and 1.97 respectively. We were 2nd out of 4 submissions to this task. The reason for being better than the baseline is rather simple: We exploit a large amount of data and use robust multi-stage training mechanisms.

We did not utilize large monolingual corpora for back-translation and instead focused on small in-domain corpora in order to avoid problems related to balancing large and extremely small (relatively speaking) corpora. Furthermore, we realized that it should be possible to fine-tune our models on Japanese–Russian data in order to obtain additional BLEU gains. We will pursue the use of larger monolingual data and additional fine-tuning in the future.

For additional results using other metrics, human as well as automatic, we refer the reader to the official website[6][7].

5 Tamil↔English Task

For Tamil↔English translation we used a simple fine-tuning based approach which manages to yield translations of reasonably good quality.

[5] https://github.com/taku910/mecab

[6] http://lotus.kuee.kyoto-u.ac.jp/WAT/evaluation/list.php?t=66&o=4

[7] http://lotus.kuee.kyoto-u.ac.jp/WAT/evaluation/list.php?t=67&o=1

Dataset	Sentences	English tokens	Tamil tokens
train	166,871	3,913,541	2,727,174
test	2,000	47,144	32,847
development	1,000	23,353	16,376
total	169,871	3,984,038	2,776,397
Domain	**Sentences**	**English tokens**	**Tamil tokens**
bible	26,792 (15.77%)	703,838	373,082
cinema	30,242 (17.80%)	445,230	298,419
news	112,837 (66.43%)	2,834,970	2,104,896
total	169,871	3,984,038	2,776,397

Table 2: Statistics on our in-domain parallel data for the Tamil–English task.

5.1 Datasets

The Tamil–English parallel corpus (Ramasamy et al., 2012) belongs to a mixed domain of bible, cinema and news. The corpora statistics and splits at the sentence and domain level are are described in Table 2. Additionally, we used the IITB Hindi–English parallel corpus for transfer learning via fine-tuning. This corpus consists of 1,561,840 lines. We do not use Hindi–English development set for tuning as we we pre-train for a fixed number of iterations.

5.2 Results

For Tamil→English translation we obtained a BLEU score of 27.81 which is approximately 2 BLEU below the best system wheres for the opposite direction we obtained a BLEU score of 12.74 which is only 0.31 BLEU below the best system. In the latter case, the difference is not statistically significant. Furthermore, for English→Tamil, we observed that we can obtain a statistically significant improvement over a baseline model that uses only the English–Tamil parallel corpus. We believe that this improvement comes from the strengthened encoder which is pretrained on the English–Hindi data. However, the improvement for the reverse direction using the same type of pretraining is approximately 3.5 BLEU. As such, we can conclude that fine-tuning a pre-trained model is more valuable when the target language is the same as compared to when the source language is the same. For Tamil→English our submission was ranked 3rd out of 7 submissions whereas our English→Tamil submission was ranked 2nd out of 6 submissions. In the future we will experiment with back-translation as well as mechanisms to improve the quality of transfer learning by fine-tuning. Perhaps, pre-training with multiple language pairs might give better results similar to what we observed when working on our Russian↔Japanese submission.

For additional results using other metrics, human as well as automatic, we refer the reader to the official website[8][9].

6 Conclusion

In this paper we have described our submissions to WAT 2019. We focused on multilingualism, transfer learning and back-translation for our submissions. For Russian↔Japanese we observed that our work on multilingual multi-stage training in conjunction with back-translating in-domain corpora leads to a competitive submission. On the other hand, for our Tamil↔English submissions we showed that simple transfer learning techniques such as fine-tuning can reliably improve translation quality especially for translation into English. Having noted the importance of multilingual pre-training, in the future, we will focus on fine-tuning extremely large multilingual models that use more parameters as well as layers. In particular we expect that fine-tuning multilingual BERT models (XLM) (Lample and Conneau, 2019) on parallel corpora will lead to the best translations.

Acknowledgments

We would like to thank the reviewers for their comments which helped improve the quality of this paper.

References

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2015. Neural machine translation by jointly learning to align and translate. In *Proceedings of the 3rd International Conference on Learning Representations (ICLR 2015)*, San Diego, USA. International Conference on Learning Representations.

Kyunghyun Cho, Bart van Merriënboer, Çalar Gülçehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning phrase representations using rnn encoder–decoder for statistical machine translation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1724–1734, Doha, Qatar. Association for Computational Linguistics.

[8] http://lotus.kuee.kyoto-u.ac.jp/WAT/evaluation/list.php?t=72&o=4

[9] http://lotus.kuee.kyoto-u.ac.jp/WAT/evaluation/list.php?t=73&o=7

Raj Dabre, Tetsuji Nakagawa, and Hideto Kazawa. 2017. An empirical study of language relatedness for transfer learning in neural machine translation. In *Proceedings of the 31st Pacific Asia Conference on Language, Information and Computation*, pages 282–286. The National University (Phillippines).

Aizhan Imankulova, Raj Dabre, Atsushi Fujita, and Kenji Imamura. 2019. Exploiting out-of-domain parallel data through multilingual transfer learning for low-resource neural machine translation. In *Proceedings of Machine Translation Summit XVII Volume 1: Research Track*, pages 128–139, Dublin, Ireland.

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.

Tom Kocmi and Ondrej Bojar. 2018. Trivial transfer learning for low-resource neural machine translation. In *Proceedings of the Third Conference on Machine Translation: Research Papers, WMT 2018, Belgium, Brussels, October 31 - November 1, 2018*, pages 244–252.

Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, Chris Dyer, Ondrej Bojar, Alexandra Constantin, and Evan Herbst. 2007. Moses: Open source toolkit for statistical machine translation. In *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics Companion Volume Proceedings of the Demo and Poster Sessions*, pages 177–180, Prague, Czech Republic. Association for Computational Linguistics.

Guillaume Lample and Alexis Conneau. 2019. Cross-lingual language model pretraining. *CoRR*, abs/1901.07291.

Toshiaki Nakazawa, Chenchen Ding, Raj Dabre, Hideya Mino, Isao Goto, Win Pa Pa, Nobushige Doi, Yusuke Oda, Anoop Kunchukuttan, Shantipriya Parida, Ondej Bojar, and Sadao Kurohashi. 2019. Overview of the 6th workshop on Asian translation. In *Proceedings of the 6th Workshop on Asian Translation*, Hong Kong. Association for Computational Linguistics.

Loganathan Ramasamy, Ondřej Bojar, and Zdeněk Žabokrtský. 2012. Morphological processing for english-tamil statistical machine translation. In *Proceedings of the Workshop on Machine Translation and Parsing in Indian Languages (MTPIL-2012)*, pages 113–122.

Ilya Sutskever, Oriol Vinyals, and Quoc V. Le. 2014. Sequence to sequence learning with neural networks. In *Proceedings of the 27th International Conference on Neural Information Processing Systems*, NIPS'14, pages 3104–3112, Cambridge, MA, USA. MIT Press.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz 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.

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, EMNLP 2016, Austin, Texas, USA, November 1-4, 2016*, pages 1568–1575.