Kyoto University MT System Description for IWSLT 2017

Raj Dabre[1], *Fabien Cromieres*[2] *and Sadao Kurohashi*[1]

[1] Graduate School of Informatics, Kyoto University, Kyoto, Japan
[2] Japan Science and Technology Agency, Saitama, Japan

[1] dabre@nlp.ist.i.kyoto-u.ac.jp
[2] fabien@nlp.ist.i.kyoto-u.ac.jp
[1] kuro@i.kyoto-u.ac.jp

Abstract

We describe here our Machine Translation (MT) model and the results we obtained for the IWSLT 2017 Multilingual Shared Task. Motivated by Zero Shot NMT [1] we trained a Multilingual Neural Machine Translation by combining all the training data into one single collection by appending the tokens: "$< 2xx >$" (where xx is the language code of the target language) to the source sentences in order to indicate the target language they should be translated to. We observed that even in a low resource situation we were able to get translations whose quality surpass the quality of those obtained by Phrase Based Statistical Machine Translation by several BLEU points. The most surprising result we obtained was in the zero shot setting for Dutch-German and Italian-Romanian where we observed that despite using no parallel corpora between these language pairs, the NMT model was able to translate between these languages and the translations were either as good as or better (in terms of BLEU) than the non zero resource setting. We also verify that the NMT models that use feed forward layers and self attention instead of recurrent layers are extremely fast in terms of training which is useful in a NMT experimental setting.

1. Introduction

One of the most attractive features of neural machine translation (NMT) [2, 3, 4] is that it is possible to train an end to end system without the need to deal with word alignments, translation rules and complicated decoding algorithms, which are a characteristic of statistical machine translation (SMT) systems [5]. However, it is reported that NMT works better than SMT only when there is an abundance of parallel corpora. In the case of low resource domains, vanilla NMT is either worse than or comparable to SMT, due to overfitting on the small size of parallel corpora [6].

Although PBSMT is superior to NMT in low resource situations it leads to large models (phrase and reordering tables and language models) and thus is not an attractive approach, especially because it cannot lead to the development of models that are end to end. Recently, Google's multilingual system was made available to the public which was able to perform Zero Shot translation [1]. Although, it is possible to train a multilingual NMT model using a multi encoder and decoder setup [7], such a model contains a massive number of parameters and does not enable interaction between languages by means of shared encoders and decoders. Moreover, it is clear that the basic attention based encoder-decoder model is more than capable of accommodating multiple languages while keeping the number of parameters constant. Multilingual NMT (MLNMT) models are inherently more powerful than bilingual models especially when the target language for most pairs is common.

One major problem with MLNMT models is that they take a lot of time (ranging from several days to a few weeks) to train and thus it is very difficult to test out changes in approaches. This is because the original models are recurrent which need O(N) time for encoding followed by O(N) for decoding. Recently, models that use feed forward layers instead of recurrent layers [8] were proposed which are roughly an order of magnitude faster than their recurrent predecessors. Even without ensembling, they have also been shown to surpass ensembles of recurrent models by a significant amount. In a situation where time is limited and computing power (GPUs) such models (which we abbreviate as AIAYN[1]) can be a boon. It is important to note that although we refer to AIAYN as a feed forward model, the concept of self-attention is the central aspect of the overall architecture.

Since we had limited , we decided to work with the preprocessing based approach (prepending $< 2xx >$ tokens to source sentences) to train our multilingual AIAYN model. Internally, we compared our translations against those obtained using a PBSMT model and found them to be much superior.

2. Related Work

Our work can be viewed as an extension of Google's multilingual NMT work [1] with the main difference being that we used AIAYN [8]. Although, recurrent models that use multiple encoders and decoders [7] are an option, such models contain too many parameters and take even more time to

[1] The full form is Attention Is All You Need

train than bilingual models.

3. System Description

We trained MLNMT models for both the zero shot and non zero shot settings. For our models we followed the preprocessing approach [1]. For the non zero shot setting, for each language pair (20 pairs for the all pairs setting) we prepended the source language sentence with the tokens "$< 2xx >$" where xx could be any of the language codes for the languages under consideration. Following this we simply merged the corpora. Typically, it is a standard practice to oversample the smaller corpora but since all the corpora provided, were of the same size (in terms of number of lines), we skipped this step. For the zero shot setting we simply excluded the parallel corpora for the (bidirectional) language pairs German-Dutch and Italian-Romanian. While decoding, the input sentences are prepended with the token "$< 2xx >$" in order to force the model to translate to the target language whose language code is indicated by "xx". Apart from this we made no modifications to the NMT architecture or the decoding procedure.

We also created a multilingual PBSMT model by using a simple trick. We simply prepended every token in the source language sentences with the token "xx#" where xx indicates the target language. We also trained a joint language model on a concatenated corpora of the target side of all languages. This was enough to train a single multilingual SMT model. The working of such a model is as follows: Since each source word is marked by the "xx#" token, the phrase table contains unique entries for phrases for every language pair. During testing time, to translate from Dutch to Romanian, the input sentence will contain words marked with "ro#" and this sentence will match phrase pairs that are extracted from the Dutch-Romanian parallel corpus. Despite the non standard nature of this approach, it works well in practice. Since our focus was on NMT models we did not pursue this approach further, especially because it cannot be used to perform zero shot translation.

4. Experimental Settings

We worked on training a single NMT model for all the language directions in the multilingual task. The languages involved are German, English, Romanian, Italian and Dutch for which the language codes are de, en ,ro, it and nl" respectively. English, German and Dutch are Germanic languages whereas Romanian and Italian are Romance languages. Since they are all European languages and share cognates and grammatical structure, a multilingual model by means of parameter sharing can benefit greatly due to the language similarity.

For our experiments we used the parallel corpora provided to us by the organizers. For the non zero shot setting there are 20 parallel corpora for each language direction (5 languages and 4 targets per language leading to 20 pairs). For the zero shot setting (where the Italian-Romanian and German-Dutch corpora were to be excluded) we used only 16 out of the 20 parallel corpora. Kindly refer to the workshop overview paper for details on sizes. Apart from the official test set for this year's shared task we also evaluated our models using the "tst2010" test set that was provided to us along with the training data. Since the training, development and test sets are available in xml format we did preprocessing in the following order[2]:

1. Remove all XML tags so as to leave only raw sentences
2. Tokenize using the tokenizer in Moses[3].
3. Learn and apply a truecaser model[4] which deals with capitalization.
4. Optional 1: For the PBSMT models learn and apply a joint BPE model[5] to reduce data sparsity.

Following these steps we performed the following preprocessing steps to enable multilingual translations in a black box setting. For the PBSMT model we prepended every source language word with the token "xx#" corresponding to the target language. For the NMT models we prepended each source language sentence with the token "$< 2xx >$".

For training we used Moses[6] for the PBSMT model and Tensor2Tensor's implementation of AIAYN[7] for the NMT model.

For PBSMT the settings are:

- Subword vocabulary size of 32000 before appending the "xx#" tokens.
- A joint 7 gram KenLM model[8] [9] to account
- Default training settings for the phrase tables.
- Default settings for tuning using MIRA via MERT.

For NMT the settings are:

- Subword vocabulary size of 32000 which the subword tokenizer in the AIAYN implementation generates automatically.
- Embeddings and layer outputs of sizes 512 and the feed forward layer with a hidden later size of 2048.

[2] To generate the submission files we simply undid the preprocessing in the reverse direction

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

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

[5] https://github.com/rsennrich/subword-nmt

[6] https://github.com/moses-smt/mosesdecoder

[7] https://github.com/tensorflow/tensor2tensor/tree/master/tensor2tensor

[8] https://github.com/kpu/kenlm

L1/L2	de	en	it	nl	ro
de	-	26.45	17.54	19.64	16.27
en	23.25	-	30.79	28.80	24.66
it	19.10	34.73	-	22.32	20.60
nl	20.27	30.49	19.86	-	17.65
ro	17.94	29.58	21.89	20.24	-

Table 1: The official evaluation results for the multilingual NMT model task (non zero shot case).

L1/L2	de	en	it	nl	ro
de	-	27.08	17.67	**20.31**	16.08
en	23.63	-	30.99	30.18	24.49
it	19.20	35.28	-	22.76	**20.37**
nl	**19.68**	30.63	20.74	-	17.74
ro	18.40	30.23	**21.85**	20.47	-

Table 2: The official evaluation results for the multilingual NMT model task (zero shot case). The results for the zero shot pairs are marked in bold.

- Adam optimizer with a weight decay on the learning rate that increases for 16000 iterations and then decreases.
- Beam of size 4 with an alpha value of 0.6 for decoding the test sets.

We trained our models for 400000 iterations which is equivalent to roughly 10 epochs that required only 3-4 days on 5 GPUs. With 8 GPUs which is the default setting in the original AIAYN paper we can expect faster convergence. We did experience a slight amount of overfitting and could have eliminated it with dropout but will pursue such activities in the future. We also did not average the model checkpoints before decoding and instead only took the final model[9] for decoding. Decoding for all language pairs was done in parallel on multiple GPUs and took roughly an hour for all the test sets. The automatic evaluation measure we used was BLEU[10] [11] which we compute for the detokenized sentences.

5. Results

First we give the results of the official evaluation for the non zero shot and zero shot settings in Tables 1 and 2 respectively followed by the evaluations on the ”tst2010” test set which was provided along with the training data in Table 3.

L1/L2	de	en	it	nl	ro
de	-	29.63 34.98	17.57 21.37	23.51 23.69	14.49 18.96
en	21.70 27.81	-	24.04 29.07	27.25 30.91	21.38 26.65
it	15.88 21.37	28.89 34.58	-	18.48 21.83	19.46 20.72
nl	21.57 24.45	34.79 38.86	18.84 23.02	-	15.99 20.68
ro	15.96 21.81	31.10 37.10	22.65 24.07	18.57 23.01	-

Table 3: The results for the ”tst2010” set which we used as a test set for our local evaluations. Each cell contains 2 scores the one on the top is for the multilingual PBSMT system and the one on the bottom is for the multilingual NMT system.

Since we are not aware of the BLEU scores for the runs submitted by the other participants we are unable to comment on how well our results are compared to others. However, we do have interesting observations regarding our zero shot results. Despite having no parallel corpora between the Italian-Romanian and German-Dutch language pairs, the zero shot NMT model performs almost as well for translations between these pairs. For German-Dutch the non zero shot model gives a BLEU of 19.64 whereas the zero shot model gives a BLEU of 20.31 which is a significant improvement. For the reverse direction though, the non zero shot model gave a BLEU of 20.27 against 19.68 BLEU for the the zero shot model. Although, there is a drop in translation quality it is not large. For the Italian-Romanian pair (both directions) the differences between the two settings is insignificant.

Zero Shot NMT between a language pair is known to give relatively lower BLEU scores as compared to a non zero shot scenario and thus the outcomes above puzzled us initially. We decided to inspect the parallel corpora for any oddities. After some preliminary analysis we discovered that, although, the corpora are available in their bilingual form there are about 150,000 N-lingual sentences in the overall collection. For example, out of approximately 250,000 sentences for Italian-Romanian, 150,000 (60%) sentences contain translations to other languages. This means that even if the Italian-Romanian parallel corpus is excluded from the training set, there is an indirect parallel corpus of 150,000 sentences between the two languages. This also means that this setting is not truly zero shot because of the existence of the 150,000 multilingual sentences. It would be interesting to see what would happen in case all the bilingual corpora are disjoint [11].

Apart from this we also see that the zero shot models performed slightly better than the non zero shot models in a number of cases and we believe that since the non zero shot models had to work with a larger number of language pairs, the training process was no effective enough. It is possible to argue that using models with more parameters might be a good idea but we have already mentioned that our models actually overfit on the training data which means that it is better

[9] Such models overfit on the training data since they have a slightly lower BLEU on the development set than some of the past checkpoints.

[10] This is computed by the multi-bleu.pl script, which can be downloaded from the public implementation of Moses [10].

[11] In other words, these corpora come from different parts of the TED corpora with zero overlaps in their content.

consider approaches where we design better training schedules or work with better models that can incorporate multiple languages better than the kind of models we are currently using.

In Table 3 we can see how well the NMT system we trained is compared to the PBSMT system. In most cases the difference is over 4 BLEU points. The multilingual PBSMT system is simply a hack, as is the NMT system, in the sense that we only concatenated the corpora. However in the NMT system multiple languages share a common representation space which allow them to interact with each other and elevate the overall translation quality.

Although we do not mention it in the experimental section we did experiment with training a multilingual RNN model using Kyoto NMT[12] [12]. The model size was roughly the same but even after 2 weeks of training we were unable to obtain peak performance in terms of BLEU. Overall, we tried training models for about a month after which we gave up and moved over to AIAYN models and as a result were able to train high quality models within a matter of 3-4 days.

As we have mentioned our models are slightly overfitted on the training data and we also do not average various model checkpoints. We believe that the BLEU scores above can be further increased by a few points but since we were not aware of advanced techniques like model averaging and lacked the time and resources for trying out various model settings we were unable to train the best possible models. Note that we also do not do ensembling which is something that the authors of tensor2tensor do not implement and is particularly unnecessary since model averaging seems to mitigate the need for ensembling many models. We believe that in the future these AIAYN models can be exploited to their fullest extent and will replace the traditional RNN models.

6. Conclusions

We have described how we trained our zero and non zero shot multilingual NMT model for the IWSLT Multilingual MT tasks. We used the simple token based (appending "$< 2xx >$" to the source language sentence where xx is the target language) approach and observed that it is much superior to a PBSMT system. We observed that for the given corpora and settings the zero shot results are as good as the non zero shot results because of the existence of N-lingual sentences which constitute 60% of the bilingual corpora. We also verified that AIAYN models are extremely fast to train and yield models of high quality in a matter of days instead of weeks or months which the recurrent NMT models require.

7. Acknowledgements

We would like to thank the creators of tensor2tensor for making their code available since it allowed us to conduct several NMT experiments in a few days which would have required weeks, if not months, had we relied on recurrent NMT models. We would also like to thank the organizers and the anonymous reviewers for their efforts. We would also like to thank MEXT (Japan) since their scholarship is the source of funding for the first author.

8. References

[1] M. Johnson, M. Schuster, Q. V. Le, M. Krikun, Y. Wu, Z. Chen, N. Thorat, F. B. Vigas, M. Wattenberg, G. Corrado, M. Hughes, and J. Dean, "Google's multilingual neural machine translation system: Enabling zero-shot translation." *CoRR*, vol. abs/1611.04558, 2016. [Online]. Available: http://dblp.uni-trier.de/db/journals/corr/corr1611.html

[2] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," in *In Proceedings of the 3rd International Conference on Learning Representations (ICLR 2015).* San Diego, USA: International Conference on Learning Representations, May 2015.

[3] K. Cho, B. van Merriënboer, Ç. Gülçehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, "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).* Doha, Qatar: Association for Computational Linguistics, Oct. 2014, pp. 1724–1734. [Online]. Available: http://www.aclweb.org/anthology/D14-1179

[4] I. Sutskever, O. Vinyals, and Q. V. Le, "Sequence to sequence learning with neural networks," in *Proceedings of the 27th International Conference on Neural Information Processing Systems*, ser. NIPS'14. Cambridge, MA, USA: MIT Press, 2014, pp. 3104–3112. [Online]. Available: http://dl.acm.org/citation.cfm?id=2969033.2969173

[5] P. Koehn, H. Hoang, A. Birch, C. Callison-Burch, M. Federico, N. Bertoldi, B. Cowan, W. Shen, C. Moran, R. Zens, C. Dyer, O. Bojar, A. Constantin, and E. Herbst, "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.* Prague, Czech Republic: Association for Computational Linguistics, June 2007, pp. 177–180. [Online]. Available: http://www.aclweb.org/anthology/P/P07/P07-2045

[6] B. Zoph, D. Yuret, J. May, and K. Knight, "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,*

[12] https://github.com/fabiencro/knmt

2016, 2016, pp. 1568–1575. [Online]. Available: http://aclweb.org/anthology/D/D16/D16-1163.pdf

[7] O. Firat, K. Cho, and Y. Bengio, "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*, 2016, pp. 866–875. [Online]. Available: http://aclweb.org/anthology/N/N16/N16-1101.pdf

[8] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," *CoRR*, vol. abs/1706.03762, 2017. [Online]. Available: http://arxiv.org/abs/1706.03762

[9] K. Heafield, "Kenlm: Faster and smaller language model queries," in *WMT@EMNLP*, 2011.

[10] P. Koehn, H. Hoang, A. Birch, C. Callison-Burch, M. Federico, N. Bertoldi, B. Cowan, W. Shen, C. Moran, R. Zens, C. Dyer, O. Bojar, A. Constantin, and E. Herbst, "Moses: Open source toolkit for statistical machine translation." in *ACL*. The Association for Computer Linguistics, 2007.

[11] K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu, "Bleu: A method for automatic evaluation of machine translation," in *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, ser. ACL '02. Stroudsburg, PA, USA: Association for Computational Linguistics, 2002, pp. 311–318. [Online]. Available: http://dx.doi.org/10.3115/1073083.1073135

[12] F. Cromierès, "Kyoto-nmt: a neural machine translation implementation in chainer," in *COLING*, 2016.